

# **INTERACTION INDIRECTE EN RÉALITÉ VIRTUELLE À L'AIDE D'UN MÉDIATEUR**

**THÈSE N° 3323 (2005)**

**PRÉSENTÉE À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS**

Institut des systèmes informatiques et multimédias

**SECTION D'INFORMATIQUE**

**ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE**

**POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES**

**PAR**

**Patrick LEMOINE**

DEA Instrumentation et Informatique de l'Image, Université de Franche-Comté, Besançon, France  
et de nationalité française

acceptée sur proposition du jury:

Prof. D. Thalmann, directeur de thèse

Prof. G. Abou Jaoudé, rapporteur

Dr S. Garchery, rapporteur

Dr S. Mailles, rapporteur

Dr F. Vexo, rapporteur

Lausanne, EPFL  
2005



*À mes parents, à ma mère*

*À Clara*



---

# REMERCIEMENTS

Ce mémoire de thèse présente le travail de recherche que j'ai effectué au sein du laboratoire d'Infographie (VRlab) à l'Ecole Polytechnique Fédérale de Lausanne (EPFL) sous la responsabilité du Professeur Daniel Thalmann. Je tiens à le remercier très sincèrement de m'avoir accueilli dans son équipe et m'avoir accordé sa confiance durant toutes ces années.

Je tiens aussi à exprimer ma gratitude envers toutes les personnes qui ont contribué scientifiquement et humainement à la réalisation de ce travail de recherche.

Merci, bien sûr aux membres du jury de thèse, le Prof. D. Thalmann, le Prof. G. Abou Jaoudé, la Dr. S. Mailles, le Dr. S. Garchery, le Dr. F. Vexo, pour le soin qu'ils ont apporté à l'examen de cette thèse.

Je remercie Dr. Frédéric Vexo d'avoir accepté d'amender mon document afin de l'améliorer et Mario Gutiérrez pour sa collaboration, ainsi que pour ses conseils dans l'écriture de nos articles.

J'adresse également mes remerciements à Dr. Ronan Boulic avec qui j'ai eu des discussions intéressantes.

Et enfin plus personnellement, je tiens tout particulièrement à remercier Clara Huizink, pour sa précieuse collaboration dans la mise en place de certains de mes concepts, à qui je serais éternellement reconnaissant, qui d'une flamme nouvelle a éclairé ma vie, et à qui finalement je dédie cette thèse, et donc tout mon travail.

---

---

# RESUME

Actuellement beaucoup de recherches sont faites dans le domaine des interfaces multimodales afin de permettre à l'utilisateur d'accomplir des tâches complexes de manière simple, naturelle, et rapide. Ces interfaces expertes devront être capables d'estimer les risques résultant d'une action commandée, de prévenir toute action néfaste et de suggérer éventuellement des alternatives possibles. Compte tenu, de la complexité des tâches à réaliser et de la croissance exponentielle des informations à prendre en compte, les systèmes adaptatifs sont dorénavant indispensables pour rendre possible et faciliter le travail de l'opérateur. Une bonne interface homme machine est donc dorénavant exigée.

Nous constatons, que de multiples techniques d'interactions et de manipulations sont actuellement disponibles sur le marché, mais que jusqu'à maintenant, les outils caractéristiques du paradigme WIMP (*Windows, Icons, Menus and Pointing device*) n'ont pas trouvé leurs homologues dans les interfaces à trois dimensions. Il reste encore du chemin à faire pour pouvoir trouver l'outil idéal et l'imposer comme un moyen standard pour les interfaces et applications 3D.

C'est pourquoi, nos recherches se sont focalisées progressivement vers la proposition d'une « **interface médiatrice** », adaptative et très fonctionnelle, destinée à simplifier au maximum l'interaction humaine dans l'exécution de travaux complexes.

Le concept du « **médiateur** » pourrait s'expliciter de la manière suivante : Un utilisateur plongé en immersion totale dans un univers nommé *monde médiateur*, va pouvoir agir à distance, par l'intermédiaire d'un périphérique « *haptique* », sur un autre monde virtuel ou réel nommé monde contrôlé.

Rappelons que l'Homme a besoin d'outils simples pour pouvoir réaliser des tâches compliquées. Par principe, ce n'est pas à l'homme de s'adapter à la machine, mais que c'est à la machine de s'adapter à lui.

---



---

# ABSTRACT

Currently many researches in the field of multimodal interfaces (input, output) have been made in order to be able to achieve complex tasks merely, naturally, and quickly. Expert interfaces should be considering the risks resulting from an ordered action, to prevent any harmful action and to suggest possible alternatives. Taking into account the complexity of the tasks to achieve and exponential growth of information, the adaptive systems are henceforth essential to make possible and facilitate the work of the operator. A good man-machine interface is thus strongly required.

We note that multiple interaction and manipulation techniques are currently available, but at this time, the characteristic tools of the WIMP paradigm (*Windows, Icons, Menus and Pointing device*) did not find their equivalent in 3D interfaces. There still remains way to make to be able to find the perfect tool and to enforce it as a standard for the 3D interfaces and applications.

Therefore, our research was focused gradually towards the proposal for a mediating interface: a very adaptive and functional interface, intended to simplify to the maximum the human interaction in the execution of complex work.

The concept of the "mediator" might be clarified in the following way, i.e.: A user in full immersive system named mediator world will be able to control or interact a front distance, through an intermediary haptic devices, on another virtual or real world named controlled world.

Let us recall that the Human needs simple tools to be able to achieve complicated tasks. In such a case, one of the ultimate goals is to make the machine adapt to the human instead of forcing the human to adapt to the machine

---

---

# SOMMAIRE

<b>1. INTRODUCTION .....</b>	<b>15</b>
1.1. CONTRIBUTIONS.....	17
1.2. L'APPLICATIF DU MEDiateUR.....	20
1.3. PRESENTATION DU DOCUMENT .....	23
<b>2. CHAPITRE 1 : ETAT DE L'ART .....</b>	<b>25</b>
2.1. INTRODUCTION .....	25
2.2. L'INTERFACE .....	26
2.3. L'INTERACTION AVEC ENVIRONNEMENTS VIRTUELS .....	29
2.3.1. Interaction directe .....	29
2.3.2. Interaction indirecte .....	37
2.4. L'INTERACTION HAPTIQUE .....	41
2.4.1. Pour quels types d'applications .....	44
2.4.2. Etat de l'art .....	47
2.4.3. Présentation de quelques périphériques .....	53
2.5. DISCUSSION .....	57
2.5.1. Pourquoi les informations visuelles et tactiles sont importantes ? .....	58
2.5.2. Haptique contextuels passive ou active .....	59
2.5.3. Le pseudo-haptiques .....	60
2.6. CONCLUSION.....	61
<b>3. CHAPITRE 2 : PROJET VIRTUAL.....</b>	<b>63</b>
3.1. INTRODUCTION .....	63
3.2. PRESENTATION DES DIFFERENTS SYSTEMES .....	69
3.3. PRESENTATION DES MODULES COMPOSANT LES SYSTEMES .....	76
3.3.1. Le module du contrôle du système .....	77
3.3.2. Le module graphique de la scène et de visualisation .....	78
3.3.3. Le module modèle physique pour la voiture .....	81
3.3.4. Le module de gestion de collision .....	81
3.3.5. Le module de collision voiture et objets de la scène .....	84
3.3.6. Le module gérant l'avatar humain et ses collisions .....	85
3.4. DISCUSSION .....	91
3.5. CONCLUSION.....	96

---

<b>4.</b>	<b>CHAPITRE 3 : LE CONCEPT DU MEDITATEUR .....</b>	<b>99</b>
4.1.	LE CONCEPT .....	99
4.2.	JUSTIFICATION.....	104
4.3.	REFEXIONS SUR LE CONCEPT .....	107
4.4.	DISCUSSION .....	115
4.5.	CONCLUSION.....	116
<b>5.</b>	<b>CHAPITRE 4 : APPLICATION DU MEDIATEUR.....</b>	<b>119</b>
5.1.	LE MEDIATEUR DANS UN CONTEXTE DE MENUS 3D .....	120
5.1.1.	Objectifs .....	120
5.1.2.	Notre première implémentation de menus 3D .....	124
5.1.3.	Recherche d'un nouveau design de menu 3D .....	127
5.1.4.	Notre seconde implémentation : le « <i>menu pétales</i> » .....	130
5.2.	LE MEDIATEUR ET LE CONTROLE D'UN ROBOT .....	136
5.2.1.	Objectifs .....	136
5.2.2.	Présentation de l'application .....	137
5.3.	LE MEDIATEUR DANS UN CONTEXTE DE TELEREHABILITATION .....	140
5.3.1.	Objectifs .....	140
5.3.2.	Présentation de l'application .....	141
5.4.	LE MEDIATEUR DANS UN CONTEXTE DE TELEOPERATION.....	143
5.4.1.	Objectifs .....	143
5.4.2.	Présentation de l'application .....	144
5.5.	INTERFACE MEDIATRICE POUR TELECOMMANDER UN ROBOT.....	149
5.6.	CONCLUSION.....	151
<b>6.</b>	<b>CONCLUSION .....</b>	<b>153</b>
6.1.	RAPPELS.....	153
6.2.	CONTRIBUTIONS.....	155
6.3.	L' INTERFACE MEDIATRICE ET SES POSSIBILITES D'UTILISATION.....	156
6.4.	CONSTATATIONS ET RECOMMANDATIONS .....	159
6.5.	CONCLUSION.....	161
<b>7.</b>	<b>BIBLIOGRAPHIE .....</b>	<b>163</b>

# LISTE DES FIGURES

Figure 1 : Illustration de quelques méthodes d'interactions .....	32
Figure 2 : Catalogue des gestes de saisies.....	35
Figure 3 : Présentation de menus 3D.....	38
Figure 4 : Illustration de certaines techniques d'interactions indirectes.....	39
Figure 5 : Illustration d'une manipulation en tétémédecine.....	44
Figure 6 : Présentation d'un prototype d'exosquelette.....	46
Figure 7 : Un modèle de CyberGrasp.....	50
Figure 8 : Stelarc Exoskeleton 1999.....	51
Figure 9 : Illustration du principe retour « <i>haptique passif</i> » .....	59
Figure 10 : Le système 1 .....	64
Figure 11 : Le système 2 .....	65
Figure 12 : Le système 3 .....	65
Figure 13 : architecture générale du système.....	67
Figure 14 : « <b>Mock-up</b> » physiques, périphérique de contrôle.....	69
Figure 15 : Configuration « HMD » ; .....	70
Figure 17 : Les différents systèmes.....	72
Figure 18 : Organigramme des différents systèmes .....	74
Figure 19 : Diagramme simplifié.....	76
Figure 20 : Comment fonctionne la stéréovision ?.....	78
Figure 21 : Visualisation des problèmes de projection sur les murs.....	79
Figure 22 : Schématisation du système CAVE.....	80
Figure 23 : Le système CAVE.....	80
Figure 24 : Regroupement par paires de triangles.....	84
Figure 25 : Repositionnement des senseurs autour de l'objet .....	88
Figure 29 : Calcul des collisions entre l'avatar piloté par l'utilisateur et les objets primitifs .....	91
Figure 30 : Conduite de la voiture sous HMD.....	92
Figure 31 : Exosquelette de PERCRO ; Le WHIPFI prototype du CEA ; La « <b>station haptique</b> » d'IMMERSION™.....	95
Figure 32 : Vues de VIRTUAL.....	98
Figure 33 : Illustration de piétons marchant dans une ville.....	98
Figure 34 : Représentation schématique du concept.....	102
Figure 36 : Schéma simplifié du concept.....	111
Figure 37 : Schéma plus détaillé du concept .....	112
Figure 39 : Distribution spatiales des menus.....	121
Figure 40 : Interaction avec nos menus 3D.....	125
Figure 41 : Visualisation des problèmes.....	125
Figure 42 : Différents types de design avec lesquels on est plus ou familiarisé.....	127
Figure 43 : Autre type In-"situ menu" et "slider" de control pour ajuster différents paramètres .....	127
Figure 44 : Le C3 Menu.....	129
Figure 45 : 3D Spin Menu Interface.....	129
Figure 46 : navigation au sein du « <b>menu pétales</b> » .....	130
Figure 47 : Présentation des tests accomplis par différents utilisateurs dans notre monde virtuel.....	133
Figure 48 : Présentation du système pour contrôler un robot .....	137
Figure 49 : vue sur les positions des membres supérieurs.....	138
Figure 50 : Parcours d'obstacles pour un robot.....	139
Figure 52 : Le thérapeute donne des directives au patient.....	142
Figure 53 : Schéma de principe .....	143
Figure 54 : « <b>haptic workstation</b> » :.....	144
Figure 55 : Illustration de l'application étudiant.....	145
Figure 56 : Illustration du système 2.....	146
Figure 57 : Les étapes de la construction de l'interface.....	148
Figure 59 : Dans le concept Téléopérateur.....	150

---

# INTRODUCTION

La finalité de cette recherche est de proposer un concept efficace permettant à un opérateur plongé dans un environnement virtuel et utilisant des périphériques « *haptiques* » à retour d'effort, de pouvoir contrôler des objets complexes, comme par exemple des robots ou des objets virtuels, le plus simplement possible.

Actuellement beaucoup de recherches sont faites dans les interfaces multimodales (input, output) afin de pouvoir accomplir des tâches complexes de manière simple, naturelle, et rapide. Ces interfaces homme machine intelligentes incluent l'analyse des entrées, la gestion d'interaction, le modelage et l'adaptation à l'utilisateur, et le support pour l'interaction avec les systèmes sous-jacents. Dans un certain nombre de domaines (« téléopération », génie logiciel), les interfaces intelligentes prennent de plus en plus d'importance. Ces interfaces devront pouvoir prévenir les conséquences néfastes d'une action ou suggérer les alternatives possibles. En plus de supporter une gamme beaucoup plus riche de styles de l'interaction, ces interfaces permettent à l'utilisateur d'exécuter des actions qu'il ne pourrait peut-être pas faire autrement. De plus, face à la complexité de certains systèmes et à la croissance de l'information, des systèmes adaptatifs sont dorénavant exigés afin de faciliter au maximum le travail de l'utilisateur.

De même dans le domaine de l'« *haptique* » nombre de recherches ont été faites et ont conduit à développer de multiples techniques, surtout dans le secteur de la « téléopération » et de la « télérobotique ». Les dispositifs « *haptiques* » sont actuellement un trait émergeant de la réalité virtuelle. Ils ouvrent des champs très prometteurs car les techniques ainsi développées sont peu à peu utilisées pour le divertissement, l'apprentissage, l'acquisition de compétences, la simulation chirurgicale, la visualisation de données scientifiques, l'assistance aux aveugles et déficients visuels, etc. De nouvelles interfaces « *haptiques* » et de nouveaux concepts apparaissent régulièrement.

Revenons maintenant à notre concept. Avant de développer de nouveaux concepts, il est primordial de se poser les bonnes questions permettant de définir les critères d'une bonne interface. La réussite ou l'échec d'un nouveau concept réside souvent sur ces bases. Une large discussion sera faite à ce sujet. La description d'une nouvelle technique est informelle et, par conséquent, très souvent partielle ce qui implique que lors de l'élaboration d'un nouveau concept d'interface, il faille garder toujours en tête, les questions suivantes : le système est acceptable mais pourquoi ? Comment l'évaluer, qualitativement, quantitativement ? Comment mesurer l'adaptabilité résultante dans le processus de conception ? De quoi s'agit-il ? (Objet, opération, nature, quantité,...) Qui est concerné ? Comment procède-t-on ? Pourquoi cela se passe-t-il ainsi ? (Les causes, raisons, etc..).

Nous allons donc, tout au long de cette recherche, par le biais de tests effectués sur des applications concrètes telles que le projet européen VIRTUAL (simulateur de voiture doté de plusieurs interfaces de commandes), constater les principales difficultés rencontrées tant au niveau des concepts que de l'interaction et comprendre ce qui nous a conduit à passer de l'interaction directe au concept du médiateur.

Nous pouvons constater que certaines manipulations d'entités (objets virtuelles, robots,...) requièrent l'utilisation d'intermédiaires, simplifiant ainsi l'interaction l'homme et l'entité désignés sous le vocable « médiateurs ». Ce sont des outils simples, facilitant l'interaction s'ils sont bien adaptés. Pour bien comprendre cette notion de médiateur nous pouvons citer les exemples suivants :



- Prenons l'exemple le plus courant. Nous avons une personne qui parle une langue A et l'autre une langue B. Puisqu'un dialogue ne peut pas s'établir entre ces deux personnes, il faut faire intervenir une tierce personne connaissant les deux langues. Donc, un traducteur qui joue le rôle de médiateur entre les deux personnes A et B, permettant d'établir la communication.
- Prenons l'exemple de la photo ci-contre du robot et de l'assistante (Figure : Honda Motor Co - P2 Humanoid : Asimo (1.2m x 0.45m x 0.44m, 43 kg, 26 DOF)). Une



personne (A) par l'intermédiaire du robot va dire bonjour en tendant la main à l'assistante (B) qui vient vers lui. Le robot est un médiateur entre l'opérateur et l'assistante.

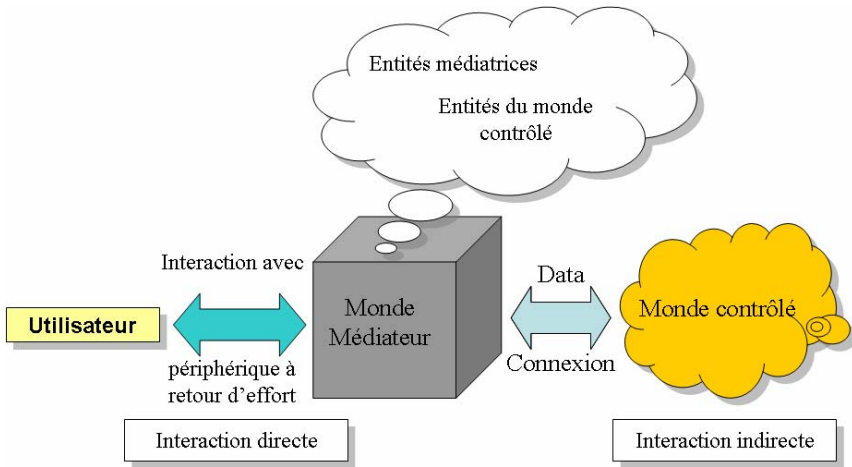
- Autre exemple, celui du musicien. Il doit produire un son, mais il ne peut pas le faire directement. Il utilise pour cela un instrument de musique qui est médiateur entre le musicien et le son produit.

Finalement, c'est ce concept là que nous souhaitons retranscrire dans le domaine de la « téléopération » combiné à celui de la réalité virtuelle. C'est-à-dire créer une interface médiatrice entre l'humain et l'entité qu'il désire manipuler.

## 1.1. CONTRIBUTIONS

Avant d'arriver à une solution convenable, il tenait en premier lieu de rappeler tout au long de cette recherche un certain nombre de notions, ne serait ce que l'interfaçage homme machine, « l'affordance », etc.. Une multitude de questions apparaîtront, nous essaierons par nos réponses ainsi que ceux venant des travaux réalisés par les chercheurs du monde entier d'y apporter des réponses.

Donc, si nous devons définir le concept de « Médiateur » de manière générale, elle serait la suivante un utilisateur plongé en immersion totale que l'on nommera ***monde médiateur***, doté d'une « multimodalité », va pouvoir agir à distance, par l'intermédiaire d'un périphérique « ***haptique*** », sur un autre monde virtuel ou réel que l'on nommera ***monde contrôlé***. De ce fait, notre étude se concentre surtout, sur l'interaction et sur la conception de nouvelles modalités d'interaction multimodale qui aideront l'utilisateur à appréhender, à gérer, à interagir avec des entités du monde contrôlé. Nous vous présentons ici, sous forme simplifiée notre concept du médiateur.



- **Le monde contrôlé** : c'est un monde virtuel ou réel où des entités évoluent et peuvent être modifiées. Dans notre réalité quotidienne, ces entités peuvent être des robots, des machines comme dans la téléopération, ou de la méta-information.
- **Le monde médiateur** : c'est un monde virtuel contenant des outils définis par l'utilisateur suivant ses besoins et permettant d'interagir sur les entités du monde contrôlé.
- **Une entité** : c'est, dans le monde médiateur, toute représentation symbolique des objets du monde contrôlé, ainsi que les outils permettant de les contrôler. Une entité possède donc un ensemble de capteurs et d'effecteurs au sens théorique du terme qui lui permettent d'interagir avec son environnement.

Nous allons plutôt nous focaliser sur l'un des aspects les plus concrets observé dans les univers de la « téléopération » et de la robotique. Dans cette étude, les simulations traitées sont pour la plupart virtuelles. Le monde contrôlé sera virtuel afin d'évaluer le concept avant de le valider et de l'exporter sur le terrain opérationnel (Contrôle d'un vrai robot). L'interface de contrôle dans le secteur de la « téléopération » étant peu modulable, nous avons décidé de projeter cette interface physique de contrôle dans un monde virtuel. En effet, dans ce monde

médiateur, nous avons la totale liberté de pouvoir créer n'importe quelle interface de commande virtuelle. Après avoir défini les modalités de sa construction, il faudra réaliser des outils permettant à l'utilisateur de pouvoir construire sa propre interface de contrôle par programmation visuelle. L'utilisateur pourra ainsi placer les entités médiatrices dont il aura besoin dans son monde médiateur. Mais cela ne sera pas suffisant, car l'utilisateur doit pouvoir mettre en relation l'entité médiatrice et l'entité du monde contrôlé. Il faudra donc définir par quels mécanismes nous pourrions le faire. Une description sémantique de chaque entité pourrait, peut-être permettre de définir les règles de connexion ainsi que l'utilisation d'entité opérateur mettant en relation les deux entités (médiateur et celui du monde contrôlé).

Nous allons aussi dégager, au cours de notre recherche, toute une réflexion sur ces différents thèmes (interaction, « *haptique* », interface, « téléopération »,...) afin de déterminer les meilleures stratégies permettant d'élaborer et de concevoir une interface médiatrice hautement configurable et conviviale.

Les principales contributions de cette recherche sont :

- De faire quelques rappels sur les interfaces Homme-Machine, sur les techniques d'interactions directes et indirectes permettant de manipuler des objets.
- De développer de manière théorique, un concept adaptable selon le profil de l'utilisateur afin de permettre la création de nouveaux moyens de communication, rapides, efficaces et synthétiques au niveau de l'information.
- De fournir des outils permettant à l'utilisateur de construire sa propre interface de contrôle, et de définir les critères lui permettant d'interagir efficacement, directement ou indirectement, et de manière conviviale sur des entités. Mais également de définir les outils me permettant de construire, de programmer cette interface de contrôle, de définir le protocole de connexion des entités, de gérer le flux d'informations que l'utilisateur perçoit (visuelle, sonore ou tactile).

## 1.2. L'APPLICATIF DU MEDiateUR

Nous vous présentons ici une ébauche d'idées sur le potentiel applicatif du concept du médiateur dans la vie courante. La mise en application de ce concept devrait donc offrir des solutions qui simplifieront la vie des personnes et du point de vue économique, de réduire les coûts de développements grâce à la flexibilité des interfaces virtuelles. Nous pensons que l'utilisation d'un environnement immersif et d'une « *station haptique* » avec retour d'effort se justifie.

**Interface médiatrice pour contrôler un avatar humain ou un robot :** Prenons l'exemple d'un avatar humain virtuel soumis à une force de gravitation. L'opérateur désire lui faire changer de posture sans que l'avatar perde l'équilibre. Il a donc le choix de tirer sur les membres de l'avatar à l'aide de la souris, mais il ne ressent rien. L'avatar peut perdre à tout moment l'équilibre si la traction est trop forte. En utilisant un périphérique à retour d'effort l'utilisateur sait qu'il ne peut pas dépasser un certain seuil car son mouvement est bloqué. Certes dans le cas de l'utilisation d'une souris nous pourrions toujours afficher de l'information mais cela ne serait pas très convivial, et de plus, il y aura trop de paramètres à gérer. C'est pourquoi ressentir la perte d'équilibre d'un avatar humain en temps réel sous la forme de forces qui contraignent le mouvement de l'opérateur est une approche beaucoup plus réaliste.

Maintenant, dans le cas où notre avatar humain virtuel devient un robot humanoïde réel comme celui proposé par Honda. L'utilisateur via notre interface médiateur va pouvoir, tel un marionnettiste, contrôler ce robot et éviter les dégâts occasionnés par sa chute lors d'un déplacement en milieu accidenté. Par ce principe de commandes gestuelles avec contrainte de retour d'effort, l'utilisateur pourra gouverner son robot de manière conviviale au lieu de d'avoir à lire une pléthore d'informations non gérables en temps réel.

**Interface médiatrice multimission pour contrôler des drones :** Utilisation d'une interface médiatrice multimission pour pouvoir commander et gérer une flottille de drones semi-autonome. Un drone est un véhicule robotisé capable de mener une mission de façon plus ou moins autonome. Engins volants de taille réduite, utilisé



pour la reconnaissance, moins chers et plus simples à mettre en œuvre qu'un avion. Ses missions sont la surveillance, et la reconnaissance. Avec des objectifs multiples comme par exemple récolter des données sur le trafic, pour la prévision météorologique, utiliser pour des opérations de recherches aériennes et de sauvetage...

La prévention par les drones de feux ou...



Dans le cadre de la surveillance des feux de forêt ou du contrôle de son évolution, il serait intéressant de pouvoir gérer tous cela depuis un seul poste de commande configurable par l'utilisateur. L'opérateur en immersion totale pourrait via notre

interface contrôler une multitude de drone. Il pourrait donner des instructions de mission à chacun d'eux (trajectoire, filmer une zone d'intérêt, ...), prendre le contrôle de chacun d'eux, pouvoir les piloter via l'utilisation de périphérique virtuel de commande avec utilisation d'une « *station haptique* » pour générer les retours d'effort nécessaire. Afficher les informations utiles pour l'opérateur dans son environnement virtuel médiateur. Comme nous l'avons déjà évoqué auparavant, l'opérateur peut définir ce qu'il veut dans son espace. Il peut visualiser l'information, agir avec les entités qu'il désire contrôler. L'utilisation d'une « *station haptique* » permet de communiquer des informations sous forme de forces qui peuvent contraindre les mouvements de l'utilisateur, voire même servir comme moyen d'alerte ou de vigilance.

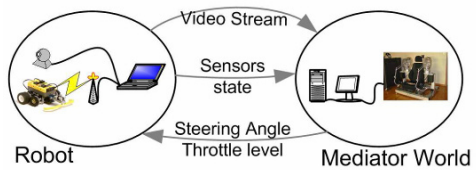
**Interface médiatrice pour contrôler des données ou des flux de données :** Là, nous entrons dans un contexte un peu plus abstrait. L'idée est de construire une interface pour pouvoir manipuler, gérer, visualiser de l'information. Le monde médiateur offre énormément d'espace sous réserve que l'on gère efficacement la répartition visuelle de l'information. Le monde contrôlé devient entièrement abstrait et immatériel car il est constitué par des informations provenant de base de données. Des outils de représentation de ces données sont nécessaires au sein de notre interface médiatrice. Un formalisme symbolique constituant une synthèse d'information est indispensable qui serait présentée sous la forme d'entité idiomatique.

Par exemple, deux entités idiomatiques A et B qui sont visibles par un opérateur qui veut pouvoir fusionner A et B afin de créer une nouvelle entité C. Pour cela l'utilisateur doit pouvoir utiliser une entité médiateur c'est-à-dire un outil mettant en relation A et B pour créer l'objet C. Nous ne faisons là que de la soit disant programmation visuelle. Nous pouvons utiliser ce principe pour fusionner à distance des bases de données compatibles. Ce qui implique qu'une description sémantique de ces deux entités est nécessaire pour pouvoir effectuer convenablement des opérations.

Comme autres applications, cet outil peut permettre de gérer et de visualiser les structures réseaux (network, téléphoniques, autre type de télécommunication, trafic routier ou électrique et intervenir dessus au cas il aurait un problème.). On peut même imaginer la situation suivante : un opérateur surveille le réseau du métro ; une alerte à lieu ; l'opérateur sélectionne un des métros sans pilote en question ; l'interface de commande du métro s'affiche devant lui (Webcam, manette de contrôle, information relatif au métro) ; il prend acte des informations et peut d'un coup de clic intervenir sur le réseau du métro tout entier. L'opérateur peut donc régler la situation à distance.

### Interface médiatrice pour contrôler

**un robot à distance :** Nous sommes ici dans le cadre de la « téléopération ». Au lieu d'avoir une interface physique pour contrôler le robot, l'interface de contrôle a été



traduite dans un environnement virtuel. Le monde médiateur contient les commandes, outils médiateurs (« *joystick* »), pour contrôler le robot du monde contrôlé qui lui est réel. [Gutierrez-1-2004]

### 1.3. PRESENTATION DU DOCUMENT

Les chapitres sont organisés de la façon suivante :

**Chapitre 1** : Présentation des outils, des concepts utilisés qui ont servi de base pour le développement de l'idée du médiateur, c'est-à-dire la définition d'une interface, de l'« *haptique* », des techniques d'interactions directes et indirectes permettant à un utilisateur d'interagir dans son environnement. Les remarques ainsi dégagées serviront de base pour le développement d'une bonne interface médiatrice.

**Chapitre 2** : Présentation du projet européen VIRTUAL (un simulateur de conduite) et des points délicats auxquels nous avons été confrontés concernant la construction des différentes interfaces, l'interaction réelle et virtuelle, les problèmes relatifs à l'« *haptique* », la détection des collisions, les synchronisations de l'application au niveau réseau et à l'implémentation du moteur dynamique pour la voiture. Les résultats ainsi obtenus, nous ont donné des éléments non négligeables pour la conceptualisation et l'élaboration de notre interface médiatrice.

**Chapitre 3** : Présentation du concept du médiateur. Nous allons faire dans cette partie une réflexion sur l'interface idéale pour pouvoir interagir de manière efficace avec des objets dits complexes. L'objectif est ici de proposer une structure générale de cette interface médiatrice, c'est-à-dire le moteur de notre concept qui gèrera tous les mécanismes dont nous aurons besoin. Nous répondrons aussi aux questions suivantes : Comment construire et gérer l'environnement du médiateur ? Comment présenter les informations à l'utilisateur ?

**Chapitre 4** : Illustration du concept du médiateur par des exemples concrets. Ces petites applications que nous allons présenter ont été les bases de notre réflexion. Nous avons pu grâce à ces applications concrétiser les difficultés rencontrées tant au niveau de l'interaction qu'au niveau de l'interface. Les résultats obtenus, nous ont permis de déterminer ce qu'il fallait faire ou non, avec le souci majeur de trouver l'interface médiatrice idéale.

**Chapitre 5** : Conclusion et perspective de recherches. Nous faisons ici un bilan du médiateur et donnons quelques recommandations pour le futur.

Maintenant nous allons décrire ce qui nous a amené au concept du médiateur, en faire sa description puis définir ses extensions possibles.





# CHAPITRE 1 : ETAT DE L'ART

L'objectif de ce chapitre est de faire un tour d'horizon sur les outils, les concepts utilisés qui ont servi de base pour le développement de l'idée du médiateur. Nous allons donc décrire ce que l'on attend d'une interface en général, puis nous nous focaliserons sur le domaine de l'« *haptique* ». Nous ferons un état de l'art sur les techniques d'interactions ainsi que sur les techniques liées à l'« *haptique* ». Les remarques exposées tout au long de ce chapitre, seront ainsi les bases du développement de notre interface médiatrice.

## 2.1. INTRODUCTION

L'évolution sans cesse croissante de la technologie dans le milieu de la réalité virtuelle permet de simuler des mondes virtuels de plus en plus réalistes au travers de logiciels et d'interfaces matérielles. La réalité virtuelle est l'art permettant de restituer la notion de réalité à un humain alors qu'elle n'existe pas dans le concret. C'est l'art d'appréhender un monde abstrait. Elle regroupe l'ensemble de techniques permettant de reproduire le plus fidèlement possible, par un calcul en temps réel, le comportement d'entités 3D en interaction entre elles. L'utilisateur dans ce monde a la sensation d'être en effet en interaction avec cet environnement virtuel. Il lui arrivera d'oublier pendant un laps de temps, que c'est seulement une simulation. Cet oubli dépend du niveau d'immersion et de présence auxquelles il est soumis. Pour cela le niveau d'immersion est souvent rehaussé à travers l'usage de matériels spécifiques, comme l'utilisation de casque de réalité virtuelle, ou de gant de données. D'autres techniques sont aussi utilisées pour accroître les simulations en réalité virtuelle, comme par exemple en rehaussant la qualité de la scène 3D ou en utilisant des techniques de visualisation 3D (la stéréoscopie). Les travaux réalisés dans le domaine de la VR ont souvent été dirigés par l'attrait technologique et non par les besoins et les objectifs humains. C'est un domaine qui évolue rapidement en parallèle avec les avancées technologiques.

Aujourd'hui face au progrès constant des technologies tant au niveau software et hardware (accélérateurs graphiques et ...), elles permettront aux utilisateurs d'interagir de

plus en plus avec des données visuelles en trois dimensions combinées à la technologie de l'« *haptique* ». Par exemple, les utilisateurs de la conception assistée par ordinateur et de visualisation de données, sont très intéressés par la possibilité de toucher et de manipuler des objets réels ou virtuels (proches ou distants). Les environnements virtuels permettent de conceptualiser de puissants outils d'interaction avec les divers mondes 3D qui sont plus ou moins une copie de notre monde réel. Les programmes développés pour ces environnements virtuels nécessitent donc de nouvelles interfaces, adaptées à leur nature spécifique. Qui dit interface, dit interaction. L'interaction humaine dépend de plus en plus des technologies de l'« *haptique* ». Ces technologies développées allègent significativement les problèmes d'interaction entre l'humain et son univers.

## 2.2. L'INTERFACE

Avant toute conceptualisation, nous devons définir ce qu'est une interface ainsi que les critères à prendre en compte pour l'élaborer correctement. Nous allons aborder ces points et les prendre en considération afin d'éviter que nous fassions les mêmes erreurs que ceux évoquées par d'autres chercheurs. Ultérieurement, nous verrons que de nombreux problèmes sont susceptibles d'altérer la qualité des interfaces. Mais un leitmotiv d'une interface réussie est sa convivialité

Revenons à la définition de ce qu'est une interface. Une interface est l'intermédiaire entre des actions voulues par l'utilisateur (représentation mentale) pour réaliser sa tâche et la technologie mise à sa disposition. C'est la représentation des actions offertes par les moyens qui influenceront l'aspect qualitatif de l'interface. Cette interface doit fournir à l'utilisateur les fonctions nécessaires dont il a besoin pour que l'utilisateur puisse mener à bien la tâche qu'il souhaiterait accomplir. Tâche n'impliquant pas, de façon contraignante, des actions étrangères à la nature de la tâche. C'est aussi une sorte de langage avec des propriétés propres à chacun des niveaux (conceptuel, sémantique, syntaxique, lexicographique et physique) qu'il faudra clairement définir.

L'utilisateur doit pouvoir modifier sa propre conceptualisation du problème pour s'adapter au mode de représentation fourni par le système. L'utilisateur se met à penser dans les termes du système et en fonction des concepts qui lui sont offerts. Ce qui comporte un danger d'appauvrissement de l'univers conceptuel de la tâche. L'intensité du sentiment d'implication

de l'utilisateur dans l'interface est aussi à prendre en compte. Permettant ainsi de réduire le sentiment de frustration, de difficulté dans les schémas de représentation mentale, et d'augmenter le caractère implicatif de l'utilisateur face à son interface. Nous savons intuitivement qu'une interface sera d'autant plus « *user-friendly* » que si elle fournit une représentation de la tâche à accomplir aussi proche que possible de la représentation mentale initiale que l'utilisateur s'en est faite. L'utilisateur ne doit pas se poser de question, la réalisation de l'opération implicite.

Lorsque l'utilisateur se trouve plongé dans un milieu immersif, les sentiments de présence et de croyance sont à prendre en considération. Il doit croire à ce qu'il voit et à ce qu'il fait pour être en adéquation avec sa représentation schématique mentale des opérations qu'il effectue ou qu'il compte effectuer. Quels sont alors les paramètres susceptibles pour obtenir une interface correcte ? Comment éviter la distorsion de l'information perçue ? Comment compenser l'influence bruitée d'un stimulus qui peut altérer le confort de l'interaction ? Comment déterminer l'impact de l'illusion ou l'influence de l'illusion dans le milieu de la réalité virtuelle ? Si nous pouvons faire croire à l'utilisateur, qu'une opération est susceptible d'avoir été faite alors qu'elle n'a pas eu lieu réellement, nous serons ainsi délivrés de certaines contraintes liées aux calculs s'exécutant en temps réel et à la qualité de l'affichage. L'utilisateur effectue alors une sorte d'interpolation ou d'illusion mentale entre deux événements lors de l'accomplissement d'une tâche.

Prenons l'exemple d'un ballon posé sur une table et qui l'instant d'après se trouve sur le sol à côté de la table. Nous sommes persuadés que le ballon est tombé, alors que visuellement nous ne l'avons pas vu tomber. Nous pouvons utiliser ces mécanismes d'illusions pour augmenter les ressources disponibles de sorte que les opérations soient exécutées rapidement, efficacement et avec le minimum de pénibilité. Nous devons réduire le fossé existant entre la projection mentale de la tâche à effectuer et la tâche réelle à effectuer.

Eviter à la personne de trop perdre du temps dans sa réflexion, c'est ce que nous devons garder en ligne de compte.

Lors de l'accomplissement d'une tâche, l'utilisateur doit passer par différentes étapes mettant en jeu des processus plus ou moins itératifs. Chaque étape se décompose en deux phases : la traduction mentale des objectifs et l'exécution de l'ordonnance des actions

**La première étape** : Traduction mentale des objectifs à atteindre.

- Etablir l'objectif à atteindre et intentions.
- Former l'intention relative à l'action qui permettra d'atteindre l'objectif. Il faudra donc d'une part déterminer par conséquent les dispositifs et mécanismes de l'interface qui produiront ces résultats et d'autre part déterminer les manipulations requises de ces mécanismes. Dans la conceptualisation de l'interface, il faudra penser et offrir cela aux utilisateurs.
- Spécifier la séquence d'actions correspondant à l'intention,

**La deuxième étape** : Exécution des actions représentées par notre modélisation mentale

- Exécuter la séquence d'actions,
- Percevoir l'état du système,
- Interpréter l'état perçu,
- Evaluer l'état interprété du système par rapport aux objectif et intention.

***Autre question, un utilisateur c'est quoi au juste ? Comment agit ou réagit-il devant une information ou une intention ?***

- exprime ses objectifs et ses intentions en des termes significatifs pour elle, en des termes liés à la représentation mentale qu'elle se fait de son problème.
- traduit ses intentions dans des actions à exécuter à l'aide des mécanismes du système interface exprimés en des termes physiques.
- perçoit physiquement le résultat des actions.
- interprète mentalement la perception physique.
- évalue la réalisation de ses objectifs à partir des variables d'état du système interface qui sont des variables physiques.

Finalement, nous devons donc prendre en compte lors de la conceptualisation de notre interface la manière dont l'ensemble des individus se comportent, tout en gardant une flexibilité et une adaptabilité suivant les goûts de chacun. Pour pouvoir interagir correctement, il doit pouvoir comprendre l'univers dans lequel il communique. Un travail au niveau de « l'affordance » est donc indispensable pour essayer de lui fournir les meilleurs outils de communications.

## 2.3. L'INTERACTION AVEC ENVIRONNEMENTS VIRTUELS

### 2.3.1. Interaction directe

Lorsqu'un utilisateur interagit dans son monde virtuel sur des objets virtuels, le moyen qui lui vient tout naturellement à l'esprit est d'utiliser ses mains. Or jusqu'à maintenant les expériences ont montré que cette manipulation d'objets en environnement virtuel est une tâche loin d'être évidente. Nous constatons que : le manque d'outils interactifs à retour d'effort, la qualité visuelle insuffisante des images ainsi et d'autres facteurs, peuvent rendre compliquée une simple tâche réelle de manipulation si elle demande beaucoup d'efforts et de dextérité.

C'est pourquoi les chercheurs dans le domaine de l'interaction ont besoin d'avoir des réponses à de nombreuses questions :

- ✓ Comment manipuler des objets sans frustration, rapidement, agilement, et avec efficacité ?
- ✓ Comment faciliter l'interaction ?
- ✓ Comment compenser les erreurs dues aux périphériques hardware ?
- ✓ Comment comparer les différentes métaphores de manipulations ainsi développées et dégager les meilleures ?

De très nombreuses techniques d'interactions dans un environnement virtuel ont été développées au cours de ces années. Ainsi, pour réaliser différentes actions l'utilisateur dispose de différents moyens d'interactions possibles :

- Les périphériques physiques isotoniques (permettant d'effectuer des tâches de pointage et de sélection d'objet avec ou sans retour d'effort)
- Les périphériques isométriques (permettant d'effectuer des translations et rotations sur l'objet sans retour « *haptique* »)
- Les périphériques avec retour d'effort adaptés à des manipulations sous contrainte
- Des outils métaphoriques d'actions et d'interactions.

### Présentation des différentes techniques d'interactions

Nous allons voir maintenant les différentes techniques de manière synthétique [Lemoine2003]. Nous distinguons trois catégories :

**“Virtual Pointer metaphor” :**

**Ray-casting** (Bolt 1980) [Bolt1980] : C’est une technique de pointage, une ligne par de la main virtuelle vers l’objet. Les objets peuvent être sélectionnés à toute distance. C’est une technique naturelle qui exige un petit effort car la sélection d’objets petits et (ou) lointains n’est pas si simple.

**Aperture** (Forsberg, 1996) [Forsberg1996] : C’est une technique basée sur le principe d’une torche qui éclaire un objet. Cette technique décrit un cône dont la source est la position de l’observateur. L’algorithme détecte alors les objets qui sont dans le volume décrit par le cône. La technique de sélection est complètement 2D et la manipulation reste encore difficile.

**Sticky Finger** (Pierce, 1997) [Pierce1997] : C’est une technique 2D basée sur le “picking” comme on le fait dans une image 2D. L’objet désigné par le doigt de l’utilisateur permet de sélectionner l’objet dans l’image 2D. Pour déterminer quel objet l’utilisateur a sélectionné, un rayon est lancé dans la scène. On regarde ce qui croise ce rayon.

**Fishing reel** (Bowman, 1997) [Bowman1997] : Cette technique est une extension du “ray-casting”. L’utilisateur peut changer la longueur du pointeur virtuel à l’aide de deux boutons. C’est une aide à la sélection des objets proches ou lointains permettant d’éviter les erreurs de sélections.

**“Virtual Hand metaphor” :**

**Go-Go** (Poupyrev, et. al. 1996) [Poupyrec1996] : Technique utilisant une métaphore de l’extension du bras de l’utilisateur. La main virtuelle va directement au contact de l’objet. L’utilisateur peut alors manipuler l’objet. Cette manipulation reste encore limitée (problèmes avec les objets lointains, saisie imprécise)

**Occlusion Selection** : La technique de la sélection par un mécanisme d'occlusion implique de placer l’extrémité du doigt entre œil et l’objet afin que le bout du doigt couvre l’objet que nous souhaitons sélectionner.

***Voodoo Dolls*** (Jeffrey S. Pierce, Brian C. Stearns, Randy Pausch) [Pierce1999] : Technique inspiré du vaudou c'est-à-dire une utilisation d'une copie pour agir sur l'objet à distance. Cette technique nécessite d'utiliser les deux mains. L'utilisateur crée une copie de manière dynamique. Une de ses mains tient la copie et l'autre permet d'agir dessus. L'avantage de cette technique c'est que l'utilisateur voit et interagit sur les deux objets en même temps. Malgré cela, nous constatons encore des problèmes de sélection, ainsi que de manipulation.

***Homer*** [Bowman1997] : C'est une technique basée sur une combinaison des techniques du RayCasting et ainsi que du Go-Go. L'utilisateur sélectionne l'objet. La main virtuelle va vers l'objet, et ramène l'objet. Les mouvements de l'objet sont liés à celle de la main. Quand l'objet est lâché, il revient à sa place, ainsi que la main. La manipulation de l'objet exige un effort beaucoup moindre.

#### **“Exocentric metaphor” :**

***Flash light*** (Liang, 1994) [Liang1994] : technique utilisant un cône de sélection plutôt qu'un rayon. La sélection des objets est plus facile à toute distance mais il reste un problème pour déterminer quel objet est sélectionné parmi ceux qui tombent dans le projecteur.

***Miniature world*** (Stoakley. Al. 1995) [Stoakley1995] : technique basé sur l'utilisation d'une vue miniaturisée de la scène virtuelle. Il y a un rapport direct entre les objets de grandeur nature dans le monde virtuel et les objets miniatures. Avec l'utilisation d'un monde miniature, l'utilisateur a le choix de sélectionner des objets en pointant sur l'objet de la scène ou sur l'objet de la scène miniaturisé. L'utilisateur peut ainsi travailler à la fois sur deux scènes identiques et d'échelles différentes. En tournant le modèle dans ses mains, l'utilisateur peut regarder et choisir les objets qu'il désire manipuler. Cette technique autorise des manipulations à toute distance. Le seul problème c'est que ce monde miniature peut occulter la scène et donc gêner l'interaction. D'une part l'affichage simultané de deux mondes peut provoquer des ralentissements et d'autre part, il est difficile de faire des manipulations d'objets dans un monde miniature.

**Scaled-world grab** (Mine , 1997) [Mine1997], [Stoakley1995] :L'utilisateur peut changer l'échelle de son monde virtuel afin de faciliter sa visualisation et donc l'interaction avec les objets du décor. La combinaison entre les actions (changement d'échelle, sélection d'un objet, et sa manipulation) est une tâche relativement fatigante pour l'utilisateur. Les interactions avec des l'objets extérieurs à la scène sont difficiles.

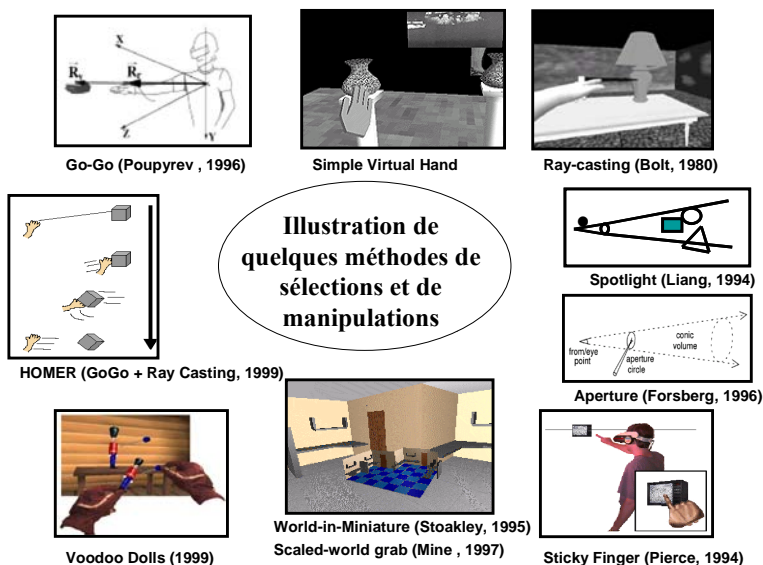


Figure 1 : Illustration de quelques méthodes d'interactions

Finalement, chacune des techniques, ainsi évoquées a son lot de points faibles ou de points forts, d'inconvénients ou d'avantages. Dans la majeure partie des cas, nous pouvons dire que la tâche de manipulation est plus difficile que celle de la sélection.

Pourquoi ? Parce que d'une part la manipulation directe en temps réel nécessite aussi beaucoup de temps de calcul en terme de collision et d'autre part, l'absence de contraintes tant au niveau sensation que visuel est un des grands problèmes rencontrés dans la manipulation d'objets virtuels.

Sans un système de contraintes physiques, l'utilisateur est cantonné à faire dans la majeure partie des cas de la manipulation approximative. Il est donc incapable d'exécuter toute sorte de manipulation précise, faute de pouvoir visualiser aisément les opérations. D'autre part, la manipulation précise exige parfois comme dans le cas de la réalité des points



d'appuis. On a besoin de créer un système de support et de contraintes pour que l'utilisateur puisse effectuer des mouvements plus précis. C'est pour cette raison que c'est une tâche difficile.

Autre remarque, les techniques fondées sur « *virtual pointer metaphor* » ne tiennent pas compte de la manipulation naturelle, à moins qu'ils soient étendus avec quelques mécanismes qui permettent à l'utilisateur de faire une manipulation correcte. Et celle de la main virtuelle est une technique plutôt intuitive parce qu'elle simule une interaction comme dans notre monde réel avec les objets. La plupart des utilisateurs novices préfèrent voir une représentation symbolique de leur main sous la forme d'une main virtuelle plutôt que de voir d'autres artéfacts car cela leur semble plus familier et moins perturbant.

### **Navigation pour aller vers l'objet**

Parmi les problèmes auxquels nous sommes confrontés, la manipulation nécessite la proximité des objets. Les objets que nous manipulons ne sont pas tous sous la main, il faut donc définir de bons paradigmes de navigation, adaptables selon les besoins de la personne. Deux possibilités s'offrent à nous, soit que l'utilisateur bouge dans l'espace réel car son espace de travail est plus ou moins limité, ce qui permet de ne pas utiliser de moyens intermédiaires pour aller vers l'objet. L'autre possibilité est d'utiliser les artifices développés dans les jeux vidéo et dans les laboratoires de recherche en réalité virtuelle. Pour résoudre ce problème d'atteinte, plusieurs techniques ont été suggérées : désigner l'objet pour aller vers lui ou désigner une direction pour permettre la navigation,... (*Voir ce qui a été fait dans les applications menus 3D*). La navigation doit être rapide et ne doit en aucun cas fatiguer l'utilisateur en augmentant la charge cognitive.

### **La sélection et utilisation des traqueurs**

L'utilisateur sélectionne avec sa main virtuelle un objet par le "toucher", et le manipule directement en déplaçant sa main (HOMER : GoGo+Ray Casting). C'est intuitif, mais limité par le sens pratique. Les utilisateurs éprouvent des difficultés comme par exemple dans la sélection de plus petits objets. L'introduction de réactions ou d'indicateurs visuelles améliore la performance pour la sélection d'objets distants et petits, mais ce n'est pas significatif pour la sélection locale. La réaction visuelle est une partie inhérente de la métaphore de la main virtuelle, mais on ne peut pas conclure si c'est un paramètre déterminant ou pas. Une des principales difficultés dans le placement des objets est liée à l'estimation des distances et la qualité du champ visuel. Certes ces méthodes d'interactions peuvent être à l'origine de

mauvais résultats. Les expériences ont surtout montré que cela vient surtout d'une part des difficultés de coordination « main-œil » et d'autre part de la qualité des mesures effectuées par les traqueurs magnétiques ou optiques. Nous constatons qu'il y a du bruit dans les mesures ce qui occasionne des problèmes de positionnement. Nous constatons que les systèmes de traqueurs infrarouges optiques donnent des mesures plus stables que les systèmes à traqueurs magnétiques. Mais ce système comporte des défauts dans l'une sont liés à des problèmes d'occultation. C'est-à-dire que la personne munie d'un de ces traqueurs peut dans ses gestes couper la liaison « caméra-traqueur ». Par conséquent une correction est nécessaire en se basant sur la connaissance des valeurs du passé, ainsi que sur l'analyse gestuelle, pour pouvoir estimer grossièrement le mouvement futur, le comparer à la mesure effectuée et le corriger. Bien entendu cette rupture de données doit être de courte durée pour donner l'illusion à l'utilisateur d'une continuité dans le mouvement.

### **Manipulation à deux mains**

Qui dit manipulation, dit utilisation des deux mains. L'interaction avec une manipulation deux mains ne frustre pas l'utilisateur dans l'accomplissement de sa tâche et n'ajoute pas de charge cognitive supplémentaire. Si nous veillons à utiliser judicieusement les deux mains en fonction de la tâche à effectuer, l'utilisateur peut accéder plus vite à l'information ou manipuler plus facilement l'objet. De plus l'utilisation des deux mains permet un repérage spatial plus aisé de l'objet placé en face de nous. La position des mains l'une par rapport à l'autre permet de mieux évaluer les distances. La deuxième main sert de référentiel, de système d'appui et d'aide pour travailler de la façon la plus naturelle possible comme nous le faisons dans la vie courante.

### **Manipulation et préhension de la main sur l'objet**

La manipulation d'objets à l'aide d'une main virtuelle nécessite des corrections en temps réel au niveau du positionnement des doigts autour de cet objet. Nous devons définir alors une tactique de correction basée sur la préhension de la main. Si nous connaissons la forme de l'objet, la disposition des doigts autour de cet objet peut ce faire très rapidement même si la disposition des doigts réels de l'utilisateur n'est pas tout à fait identique. Les gestes de l'utilisateur doivent traduire son intention de saisir l'objet et que le système s'occupe de tout. Nous devons cataloguer le geste de saisie par rapport à la forme de l'objet pour faire croire à la personne qu'elle a bien saisi l'objet. (*Voir figure 2*).

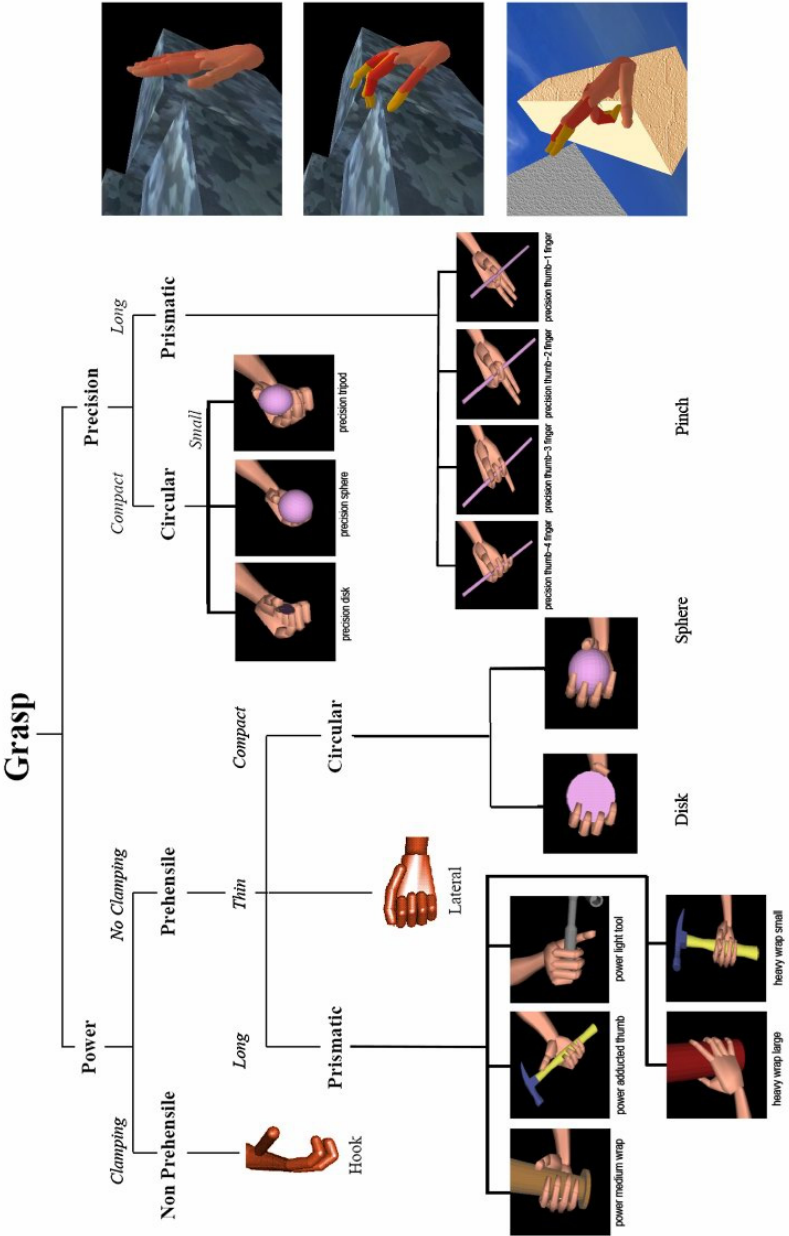


Figure 2 : Catalogue des gestes de saisies

La manipulation d'objets nécessite l'utilisation de multiples informations. Les éléments qui entrent en jeu dans la saisie d'un objet sont :

- ✓ son positionnement au sein de l'environnement virtuel,
- ✓ les caractéristiques de l'objet cible (type d'objet ; taille, forme ; la répartition de sa masse ; sa matrice d'inertie ; son coefficient de friction, sa rigidité, sa flexibilité ; ses contraintes de manipulations, etc..),
- ✓ les caractéristiques de la main (comment a t'on construit la main ; sa taille ; sa forme ; sa force ; sa dextérité, son degré de liberté pour chaque jointure, son coefficient de friction),
- ✓ les caractéristiques de l'interface (périphérique de contrôle pour pouvoir interagir ; ses contraintes d'automatisme, définir les limites de l'utilisateur ; comment l'utilisateur va interagir dans cet espace ; définir les mécanismes).

L'utilisation d'une partie de toutes ces informations, nous permet de lancer le processus de saisie. Quand un objet est considéré comme pris, la posture de la main virtuelle ne change plus. On évite ainsi que les doigts virtuels de l'utilisateur ne passent au travers de l'objet. Pour rendre la préhension plus naturelle, on adapte automatiquement la main virtuelle à l'objet qu'elle a saisi ce qui simplifie énormément les choses. C'est pour cette raison que nous devons introduire cette idée « *d'intention de* », qui consiste à étudier le mouvement gestuel de l'utilisateur par rapport à un objet cible. .

Nous devons donc anticiper les gestes, ainsi que le mouvement de « **saisie** ». Cette anticipation pourra ce faire soit par une méthode d'apprentissage avec une notion de : « *il est probable que* ». Pour prendre l'objet, il suffit alors de décrire des postures clés de saisissement.

### Manipulation et assemblages

Manipuler un objet, c'est bien mais ce n'est qu'un stade primaire. La plupart des situations ne demandent pas juste une simple manipulation d'objets (rotations ou translations sur l'objet ou sur des parties de celui-ci), mais plutôt la mise en œuvre d'activités plus complexes comme l'assemblage, démontage ou adaptation d'objets. Les interactions doivent alors être réalisées sous contraintes comme dans la réalité. Il existe des systèmes de modélisation permettant, de définir des contraintes entre points, arêtes et plans des objets, et de limiter les interactions possibles. Il y a tout de même des inconvénients à ces systèmes car

il faut définir toutes les contraintes comme on peut le faire dans ODE de Russel Smith à la construction des objets de la scène. Les contraintes doivent tout comme dans le réel être introduites dans ce monde virtuel. Les objets devront être modélisés fidèlement et nous ne devons prendre en compte que les interactions mécaniques entre objets, avec détection de collisions et gestion des propriétés mécaniques telles que le glissement. Une bonne gestion de collisions en temps réel ainsi qu'une bonne dynamique doivent être implémentées. Nous verrons ceci un peu plus loin.

## **Discussion**

Finalement, l'interaction tel que nous la concevons dans notre monde réel est très compliquée [Mark1997], parce que c'est un problème de spatialisation et qu'elle nécessite l'usage d'un bon moteur de collisions. De plus, si dans cet environnement virtuel les objets ne se comportent pas exactement comme dans le réel, il peut y avoir un conflit cognitif pour l'utilisateur. Ceci est d'autant plus important pour la construction de l'ensemble des moyens d'interactions. C'est pourquoi de nouvelles techniques sont requises. Nous devons définir de nouvelles métaphores pour l'interaction, offrir des outils adaptés et transparents pour l'utilisateur qui n'entravent en rien la compréhension de l'utilisateur. Nous devons être toujours dans l'optique d'une performance optimale avec comme critères : performance, prise en main, et utilité : est-ce que ce paradigme d'interaction aide l'utilisateur à atteindre les objectifs de sa tâche ? Ces trois points sont à prendre en considération, et sont essentiels si nous ne voulons pas que l'utilisateur soit frustré au bout de cinq minutes d'utilisation. Simplicité et performance au niveau de la productivité sont de rigueur.

Peut-être serait-il judicieux et nécessaire, dans un premier temps, d'introduire le concept de manipulation indirecte pour être capable de manipuler les objets plus confortablement. Quelques chercheurs pensent que des méthodes indirectes peuvent être assez efficaces dans les environnements virtuels bien que leurs usages enlèvent le côté du réalisme de l'environnement.

### **2.3.2. Interaction indirecte**

Nous connaissons jusqu'à présent les interfaces WIMP (**W**indows, **I**cônes, **M**enus, et **P**ointing). Nous allons voir dans cette section d'autres concepts qui ont été des sources d'inspiration pour l'élaboration de notre interface. L'objectif de cette partie est d'étudier comment interagir efficacement de manière indirecte sur des objets 3D en milieu immersif ou

« **semi-immersif** ». Afin de pouvoir proposer des outils permettant de pouvoir interagir correctement dans le cadre du concept du médiateur. Il était donc indispensable de faire un état de l'art sur les différentes techniques développées. L'interaction indirecte a nécessité beaucoup de recherches pour définir le design et les méthodes.



Figure 3 : Présentation de menus 3D

Interagir avec des objets exige des opérations de sélections permettant d'exécuter des opérations sur ces dits objets. Ceci requiert l'utilisation de menu 3D, d'objet 3D de type curseur, de widget 3D [Walker1990][Bowman1998]. La plupart des techniques ainsi utilisées dans une multitude d'applications développées jusqu'à présent reprennent les concepts déjà ainsi développés dans les interfaces et applications 2D. L'idée la plus naturelle pour les concepteurs d'interface graphique 3D a consisté à reprendre les concepts populaires et assez efficaces en 2D et de les porter directement dans le monde 3D afin que les applications proposées aux utilisateurs restent familières et intuitives. L'introduction de menus à l'intérieur d'un environnement virtuel permet à l'utilisateur de choisir une action parmi un ensemble de proposition et de réaliser des opérations pouvant être difficiles à exécuter de manière directe. Les utilisateurs sont habitués à menus standards tels que menus « **pull-down** ». De multiples projets de recherche ont proposé d'autres types de menus, d'autres ergonomies tels que « **Pie menus** », « **Marking Menus** » et... Nous les verrons dans les prochains chapitres. Dans le concept de menu 2D, le mouvement de la main de l'opérateur fait bouger un curseur ou autre chose permettant de sélectionner l'item dans un menu plan. Cette méthode d'interaction est directement analogue à l'interaction trouvée dans les postes de travail conventionnels (PC) L'avantage d'utiliser des menus, tableaux de contrôle et les boutons sont qu'ils sont familiers aux utilisateurs.

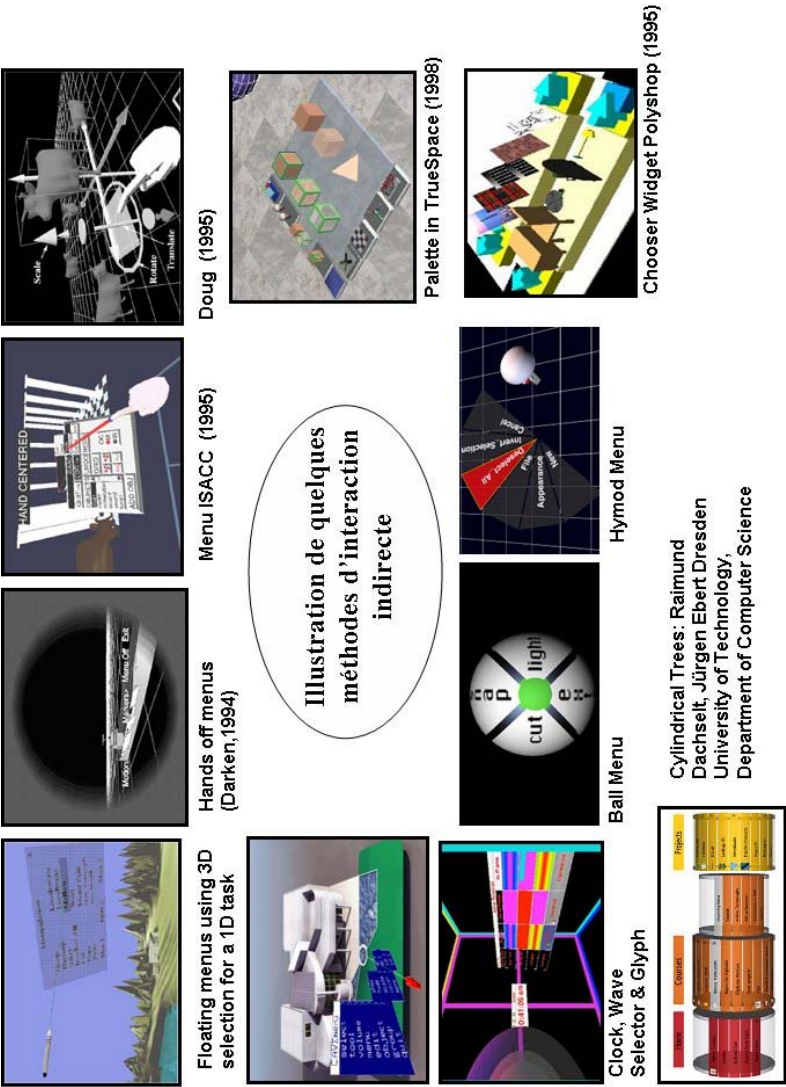


Figure 4 : Illustration de certaines techniques d'interactions indirectes.

Les menus et les « **widgets** » nous permettent d'exécuter des fonctions complexes et d'effectuer des sélectionner parmi les alternatives.

Beaucoup de chercheurs ont introduit ce concept de menus 3D à l'intérieur de l'environnement virtuel et beaucoup de recherches ont été faites au sujet des difficultés de perception par les utilisateurs afin de comprendre pourquoi les performances dans l'interaction 3D sont moindres que celles observées dans une interaction 2D.

Nous pouvons aussi remarquer que bien d'autres techniques ont été introduites pour faciliter l'interaction, comme par exemple l'utilisation « *d'outils virtuels d'interaction* ». Ce concept a été introduit par le groupe de recherche en interaction 3D de la Brown University [Conner1992] et a été intégré au système graphique UGA, dans Inventor [Strauss1992] et ...

Ce concept a été réutilisé dans les travaux de recherche de Gobbetti [Gobbetti1993] effectués dans notre laboratoire le VRlab anciennement le LIG. Ces « *outils virtuels d'interaction* », sortes de « *widget 3D* », servent à présenter une vue sélective de l'information. Ils font partie intégrante de l'environnement virtuel, pouvant être déplacés et manipulés, affectés de mécanismes de comportement permettant le contrôle, la manipulation et l'affichage des informations des objets que l'utilisateur désire manipuler.

Si on veut rentrer un peu plus dans les détails, ces outils contiennent un mécanisme, composé de nœuds d'un arbre graphique connectés par des contraintes qui définissent leur comportement. Un connecteur bidirectionnel relie certaines variables actives du mécanisme avec celles des modèles connectés. Elle offre la possibilité d'utiliser plusieurs outils en même temps pour manipuler la même information.

L'outil agit sur le « *scène graphe* » de la scène. Comme par exemple des ascenseurs, qui permettent de spécifier des valeurs réelles multidimensionnelles; des boutons, qui servent à la manipulation de valeurs booléennes; des « gadgets de connexion », « des gadgets de saisie », qui permettent de bouger des objets de manière interactive.

Finalement, l'utilisation d'une interaction indirecte pour pouvoir faciliter l'interaction est un sujet de recherche qui est loin d'être clos. Nous aborderons de manière exhaustive les techniques ainsi développées un peu plus loin.



## 2.4. L'INTERACTION HAPTIQUE

Maintenant, focalisons nous un peu plus sur l'« *haptique* ». Cette appréhension du virtuel, et de façon générale toute interface entre l'homme et la machine, passe obligatoirement par les capteurs du corps humain, à savoir les sens. Dans le cas de la manipulation, un être humain communique et agit beaucoup par ses mains, ce qui a pour conséquence de générer beaucoup d'informations.

Dans le majeur parti des cas, notre mode d'interaction ou accès aux informations reste étroitement lié au visuel. La manipulation des informations nous restreint dans la majorité des cas à utiliser le clavier ou la souris ou à des moyens hardwares peu évidents et intuitifs à mettre en œuvre. Par conséquent, il est préférable d'avoir des équivalents pour des tâches effectuées dans le milieu de l'« *haptique* » tel que pousser ou manipuler des objets virtuels ou réels. L'interfaçage « *haptique* » est connu depuis longtemps dans les applications de « téléopération » en robotique, de conduite automobile, de pilotage d'avions (joystick à retour d'effort). Ce n'est cependant qu'avec l'arrivée des techniques de réalité virtuelle que l'on a pris conscience de la problématique de « *l'interfaçage haptique* » et que l'on s'est aperçu de l'importance majeure des retours sensoriels autres que le retour visuel.

Par exemple, si un utilisateur essaie de saisir un verre virtuel, il n'y a aucun mécanisme non visuel pour empêcher la main virtuelle de traverser le verre. Si la vision permet d'interpréter une bonne partie des phénomènes, le sens « *haptique* », qui inclut la perception des efforts et du toucher, y tient probablement une place non négligeable, par le rôle qu'il joue dans toute manipulation physique.

Revenant à la définition de l'« *haptique* » : c'est un terme un utilisé par les chercheurs dans le domaine de la psychophysique humaine ayant pour objet l'étude du comment associer le sens du toucher et de la manipulation humaine avec un monde virtuel généré par ordinateur. [Salisbury1997]. La perception « *haptique* » met en jeu des phénomènes psychophysiques et physiologiques complexes qui sont encore mal compris jusqu'à maintenant. Elle permet aussi l'échange d'informations de natures physiques diverses entre l'homme et les objets de son environnement.

Le fonctionnement perceptif « *haptique* » est intimement associé à la fonction motrice chez l'être humain, et à des caractéristiques particulières qui le distinguent des autres modalités perceptives. C'est donc un lien kinesthésique entre un opérateur et un environnement virtuel. L'« *haptique* » est une technologie qui appartient au domaine des appareils interfacés avec un ordinateur. Domaine révolutionnant notre façon d'interagir avec de l'information et notre moyen communication. Ces interactions peuvent être aussi simples que toucher un mur ou un bouton qui existe uniquement dans l'esprit, ou être aussi complexes que de réaliser un acte chirurgical

La recherche en « *haptique* » essaie de résoudre par des moyens « hardware » des problèmes qui peuvent se focaliser en deux sous domaines de recherche. Celui qui traite des forces réactions (kinesthésique) et l'autre des réactions tactiles. Par abus de langage, on emploie souvent les deux termes « *retour d'effort ou retour kinesthésique* » et « *retour tactile* » au lieu du terme exact « *retour haptique* ».

Dans le premier cas, les appareils mis au point réagissent réciproquement avec les muscles et les tendons de l'utilisateur en lui appliquant des forces sur ses articulations afin de donner une existence et une consistance à l'objet virtuel manipulé. Il prend donc en charge les forces de contact, la dureté, le poids et l'inertie d'un objet. Les forces de réactions peuvent l'empêcher activement de déplacer un de ses membres dans un espace de simulation. Cependant, si nous nous restreignons à ces forces de réaction agissant sur l'utilisateur, l'information n'est pas suffisante comme dans les cas de l'information de contact d'une surface.

Dans le deuxième cas, la réaction tactile traite des appareils qui réagissent réciproquement avec les terminaisons nerveuses dans la peau (chaleur, pression, et texture). Les appareils développés sont utilisés typiquement pour indiquer si l'utilisateur est en contact ou non avec un objet virtuel et aussi pour simuler la rugosité d'un objet virtuel. Dans ce domaine, on s'intéresse moins à la force reçue qu'à sa répartition spatiale. Juste une petite remarque à ce sujet : sous la peau se trouve en effet divers capteurs en concentration plus ou moins grande. Ces capteurs permettent de détecter : chaleur, pression mécanique, frottement, orientation de la surface, vibrations, électricité.

Il faut noter que les interactions « *haptiques* » sont duales (capteur, et effecteur) et que l'établissement des modèles cognitifs de la structure physique d'un environnement provient des interactions « *haptiques* » avec les objets de notre monde.

Une chose que nous devons nous rappeler, c'est que l'« *haptique* » forme la part essentielle de nos interactions avec le monde réel. A chaque fois que nous manipulons un objet dans le monde réel, ce dernier impose des forces de réactions, de répulsion et de contact que nous percevons. Ce sont des informations kinesthésiques (forces de contact, posture et mouvement des bras et des mains par exemple). C'est grâce à cela que nous ressentons les objets. L'information tactile dit sensorielle complète alors ce manque d'information, telles que les variations spatiales et temporelles des distributions des forces des objets qui sont en contact avec la peau. Les sens kinesthésiques combinés à la capacité de mouvement nous permettent de sonder, de percevoir et de déplacer les objets de notre environnement. Cela améliore notre perception, lorsque l'on occulte l'information visuelle provenant des objets dans notre monde physique. C'est pourquoi, nous devons en tenir compte dans le cadre de la manipulation d'objet virtuel.

De plus en plus d'utilisateurs souhaitent pouvoir interagir de manière plus physique avec les mondes virtuels. La présence d'un retour d'effort en réalité virtuelle apparaît dorénavant comme nécessaire. Par exemple, dans l'industrie automobile, la maîtrise des interactions dans le cadre de l'étude de l'ergonomie du véhicule ou celui de l'optimisation des gestes de l'ouvrier dans les chaînes de montages (manipulation et assemblage des différentes pièces) est maintenant cruciale. Face à cette demande de plus en plus pressante, de multiples interfaces à retour d'effort ont été ou sont actuellement en cours de développement. Dans cette partie, nous verrons, sans trop entrer dans les détails, les multiples périphériques existants.

Le premier objectif est donc de créer l'illusion dans l'interaction, et de recréer les mêmes sensations que celles ressenties lors de la manipulation d'objet. Le second objectif est de compenser les défauts de la manipulation car manier un objet virtuel comme dans la réalité se révèle difficile.

La génération de retour « *haptique* » nécessite aussi de nombreux tests de collision entre l'objet manipulé et son environnement. Le calcul doit être rapide et doit retourner les informations nécessaires à la génération du retour « *haptique* ». Nous avons donc besoin de calculer des collisions avec précision pour pouvoir utiliser un large panel de périphériques à retour de force sachant que les périphériques de réalité virtuelle se développent et s'adaptent de plus en plus aux besoins spécifiques. Nous allons voir tout au long de cette partie ce qui a été fait dans le domaine de l'« *haptique* » et de la manipulation puis en tirer les conséquences. Est-ce vraiment une aide ou un handicap ? Devons nous utiliser de l'« *haptique* » à retour d'effort dans notre environnement médiateur ?

#### 2.4.1. Pour quels types d'applications

Ainsi toute application de réalité virtuelle qui implique une simulation du monde réel bénéficie des interactions « *haptiques* », spécialement si elles sont combinées avec des dispositifs visuels et auditifs. Le domaine de l'« *haptique* » est depuis quelques années en pleine expansion. Les systèmes à retour d'effort sont des outils permettant une meilleure adaptation aux mondes virtuels en trois dimensions. Ces périphériques sont surtout utilisés dans les simulateurs chirurgicaux, militaires, manipulation moléculaire, computer interface, design simulation, prototypage virtuel, jeux. Parmi les applications citons quelques exemples :

- Médecine : simulateurs chirurgicaux, manipulation de micro et macro robots pour la chirurgie minimale invasive, diagnostics à distance pour la télémedecine, aide pour les handicapés (ex : interfaces « *haptiques* » pour les aveugles).



Figure 5 : Illustration d'une manipulation en télémedecine

- Enseignement : donner par exemple la possibilité aux étudiants de sentir un phénomène aux échelles nanoscopiques, macroscopiques ou astronomiques. Il n'y a rien de tel de comprendre les phénomènes physiques par le ressenti avec quand même l'appui du visuel.

- Industrie : intégration de l'« *haptique* » dans les logiciels de CAO pour permettre au designer de manipuler librement les pièces d'un assemblage dans un environnement immersif. « Téléopération » (contrôle d'une machine située sur un site distant ou dans un environnement hostile).
- Art : musées dans lesquels l'utilisateur peut toucher les objets, peuvent faire leur propre sculpture mais qui reste encore très laborieux si la personne souhaite faire quelque chose de soigné.
- Jeux : jeux vidéos et simulateurs qui permettent à l'utilisateur de sentir et manipuler des solides de toute sorte. Il faut voir aussi le côté ludique de l'utilisation de l'« *haptique* » qui peut dans certain cas ne pas être une représentation fidèle de la réalité.

Dans la plupart des cas ces périphériques « *haptique* » sont utilisés dans le secteur de celui de la téléopération, télémanipulation et de la robotique. Ces systèmes ont pour principal but de conduire à distance des robots travaillant dans des endroits plus ou moins hostiles et inaccessibles pour l'opérateur humain, comme par exemple dans :

- Le domaine nucléaire : manipulation d'objets radioactifs, contrôle des tubes du circuit secondaire d'une centrale nucléaire.
- Le domaine spatial : la capture ou le dépannage de satellites
- Le domaine sous-marinier : La pose et la réparation d'installations pétrolières off-shore
- Le domaine des travaux publics : outils d'aide à la construction.
- Le domaine médical : la « téléchirurgie » pour effectuer des opérations à distance.
- Le domaine militaire : déminage, contrôle de drone pour faire des frappes chirurgicales permettant ainsi de diminuer les dommages collatéraux.

Si on veut se donner une idée de ce qu'est la « téléopération » en simplifiant extrêmement les choses, c'est une sorte de télécommande pour manipuler à distance des objets, comme par exemple un robot. Les ordres sont envoyés d'un utilisateur via une interface « *haptique* » à l'objet contrôlé. L'être humain contrôle directement en temps réel les mouvements. La tâche « téléopérante » implique aussi une réponse kinesthésique sous forme de force perçue par l'être humain. Lorsque nous manipulons des objets à l'aide d'un robot, il

est intéressant de percevoir à peu près les mêmes forces de réaction. Il y a établissement d'une relation entre l'opérateur humain et l'objet contrôlé. C'est pourquoi dans le domaine médical par exemple des efforts considérables sont faits pour essayer de reproduire au mieux l'information de la réaction sensorielle permettant d'établir un contrôle exact.

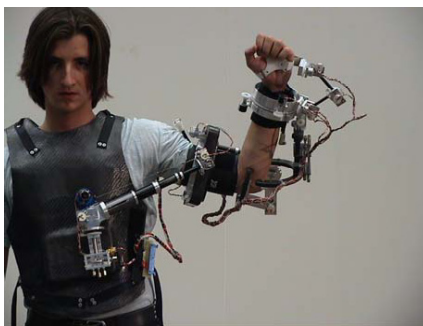


Figure 6 : Présentation d'un prototype d'exosquelette

## Discussion

Nous pouvons dire que les technologies de l'« *haptique* » sont le prochain pas important dans le développement d'environnements virtuels plus réalistes. Ces techniques offriront de grandes possibilités dans le futur.

Fournir une résistance aux objets virtuels rehausserait d'autant plus le confort dans la manipulation. Beaucoup d'applications exigent que le système soit capable de simuler en temps réel les actions et réactions. Il faut aussi que la réaction tactile atteigne le réalisme désiré avec pour objectif de percevoir les objets rigides et déformables. Nous devons savoir ce qui est exigé pour pouvoir interagir correctement, tout en gardant une certaine cohérence. Plus l'interface est assez sensible, plus le degré de spécificité de manipulation peut être élevé. Le problème avec les interfaces physiques c'est que ce sont des outils très onéreux, et difficile à reconfigurer pour faire face aux exigences des utilisateurs. L'utilisation d'une « *station haptique* » peut être une des solutions car elle élimine le caractère de spécificité de certains outils d'interaction. C'est un outil générique, offrant une grande souplesse permettant de générer, lors d'une manipulation en temps réel, les forces requises pour bloquer les membres supérieurs de l'utilisateur.

Autre point important que nous ne devons pas négliger. Les techniques d'interactions 3D doivent fournir un bon ensemble de techniques flexibles, intuitives pour les utilisateurs pour réagir réciproquement avec l'environnement virtuel. La qualité de l'interaction dépend de la simplicité, du naturel d'utilisation du système, et du caractère intuitif entre actions de l'utilisateur et les actions résultantes dans le monde virtuel. La construction d'une bonne interface « *haptique* » exige que nous prenions en compte l'utilisateur final tant au niveau de ses attentes que de ses perceptions

Aussi, le défi de toute la recherche concernant le domaine de l'« *haptique* » est de pouvoir procurer pour aux utilisateurs les même sensations que dans la réalité, de pouvoir changer la physique ou en faire une conversion à l'intérieur de son monde d'interaction (exemple traduire la chaleur par des forces de répulsions), et aussi de définir des outils permettant d'évaluer l'ergonomie du système.

#### 2.4.2. Etat de l'art

Nous allons voir maintenant un petit état de l'art de la technologie de l'« *haptique* ». Les premières études ont commencé en 1954 au laboratoire National Argonne pour le nucléaire. L'université de Caroline du Nord à Chapel-Hill à continuer à développer le prototype pour la simulation moléculaire. Ce prototype a fait l'objet d'études de 1965 jusqu'au début des années 90. A partir de là, les études de dispositifs à retour « *haptique* » se sont multipliées partout dans le monde. Ce qui a permis de confirmer qu'en « téléopération », la manipulation d'objet à distance était nettement plus efficace s'il y avait un retour d'effort.

Jusqu'à maintenant beaucoup de recherches ont été faites pour créer une interface homme machine viable pour l'utiliser efficacement dans le domaine de la réalité virtuelle ou dans les systèmes de « téléopération ». La technologie de l'« *haptique* » pour l'instant est difficile et de manière générale laisse quelque peu à désirer. Mais de constant progrès, nous permet de rester optimiste. Nous pouvons distinguer deux domaines principaux d'application et de recherche dans l'« *haptique* » : les systèmes de réalités virtuelles/télerobotique (comme par exemple les exosquelettes, « *cybergrasp* ») et à retour d'effort et tactile. Jusqu'à aujourd'hui, de nouveaux appareils d'interaction affectés à des tâches spécifiques et adaptées au monde 3D ont vu le jour.

Nous distinguons trois sous-classes différentes : ceux qui procurent une restitution graphique, ceux qui capturent les mouvements 3D, et les appareils à retour d'effort.

Comme appareils capables de capturer les mouvements 3D de l'utilisateur, nous pouvons donc citer en exemple le « *SpaceBall* » [3Dconnexion] : appareil qui fournit des données de spatialisation comme dans le cas du « joystick ». Ce périphérique est utilisé principalement dans les applications du dessin ou dans la recherche. Quant au « *Cyberglove* », développé par Virtual Technologies Corp. repris par IMMERSION™, c'est un appareil largement utilisé dans les applications de la Réalité Virtuelles. Ces gants sont généralement composés de fibres optiques ou de jauge de contrainte pour déterminer la flexion de chaque doigt. Il capture la position, donc l'orientation des différentes phalanges. Ainsi, 22-capteurs sont utilisés pour pouvoir reconstruire virtuellement la main et pour pouvoir réagir réciproquement avec les objets virtuels. Certains modèles mesurent aussi l'écartement entre chaque doigt. De plus pour déterminer complètement la position absolue de la main, il est nécessaire que les gants soient équipés de traqueurs de position dans l'espace (6 degrés de libertés).

Dans le domaine d'appareils à retour d'effort, il y a eu de nombreuses améliorations récentes. Cependant, une grande évolution technologique reste encore à faire pour obtenir un produit fini et adapté. [Burdea2003].

Les premiers modèles à apparaître ont été associés à des périphériques d'acquisition 3D dont un exemple très connu le PHANToM™ [Massie1994], composé d'un bras articulé : outil permettant de capturer les positions lors de l'interaction avec l'utilisateur, de sonder les objets virtuels et les contrôler à distance. C'est un appareil qui se déplace avec 6 degrés de liberté dans l'espace et peut résister au mouvement de l'utilisateur en fonction de sa place et son orientation. Cet appareil est principalement utilisé pour le dessin 3D mais également dans les simulateurs médicaux. Cet appareil est tout de même limité à une interaction et un ressenti ponctuel. Mais, tout de même le fait de représenter le bout d'un doigt de l'utilisateur par un unique point (le point « *ghost* »), qui nous le rappelons est en interaction avec l'environnement virtuel, accroît la résolution du sens du toucher. Quand une personne manipule un objet dans le monde réel, la résolution effective est par conséquent limitée par la taille de la surface de contact de son doigt avec l'objet.



Plusieurs appareils de type PHANToM™ ont été développés jusqu'à maintenant, et les chercheurs ont essayé d'améliorer le système. Il y a eu beaucoup de travaux avec ce type d'appareillage comme par exemple simuler un bouton physique, développer des outils virtuels pour faire de la sculpture [Thomas1996].

Gregory et al [Gregory2000] ont présenté une interface 3D pour éditer, pour modifier interactivement et peindre des mailles polygonales. Cela permet aux utilisateurs grâce à ce type de périphérique de modifier la forme d'un objet et d'éditer sa surface directement.

Mendoza et Laugier [Mendoza2001] ont présenté un modèle de la topologie local pour obtenir une force réaction stable. Leur objectif était d'utiliser une sonde virtuelle pour réagir réciproquement sur un modèle d'objet très déformable.

Tarrin et al. [Tarrin2003] ont présenté un appareil « *haptique* » à cordes où un ensemble de moteurs et ficelles des tensions des poulies pour contrôler la place d'un point d'intersection ou extrémité dans le volume visuel « semi-immersif ». L'extrémité du dispositif est attachée au doigt de l'utilisateur. L'utilisateur va pouvoir toucher par des contacts ponctuels et par l'utilisation d'un moteur de collision les objets rigides ou déformables de la scène virtuelle.

D'autres interfaces « *haptique* », en cours de perfectionnement, ont été présentées par Kawasaki et al. [Kawasaki2003], appelé Gifu qui ressemble à un membre supérieur humain dans la forme et dans les mouvements. Le bras est placé à l'opposé de la main humaine et est équipé avec les actionneurs permettant d'agir à distance en mode esclave sur des objets par l'intermédiaire d'un robot.

En plus du PHANToM™ très largement utilisé, un certain nombre d'appareils tel que les exosquelettes sont aussi disponibles et commercialisés sur le marché. Ce sont des périphériques placés sur le corps de l'utilisateur, habituellement sur ses membres sur lesquels on applique des forces. A cause de leur prix et du manque d'applications compatibles, ces appareils sont encore utilisés presque exclusivement par les centres de la recherche avancés.

Puis, il y a eu l'émergence des interfaces et applications « *multi-finger* ». Il était intéressant d'avoir des blocages au niveau des doigts. Deux types d'appareillages ont été

développés. L'un se place à l'intérieur de la main [Gomez1995] et l'autre autour de la main [Springer1997], comme c'est le cas du « **CyberGrasp** » développé par IMMERSION™. C'est un dispositif de type exosquelette, composé d'une structure métallique reliant chaque doigt par un fil. Il autorise à appliquer une force indépendante sur chaque doigt par tension exercée sur le fil correspondant.

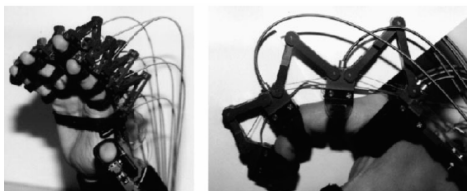


Figure 7 : Un modèle de CyberGrasp

L'utilisation du « **CyberGrasp** » seul est insuffisante car il faut pouvoir bloquer les bras de l'utilisateur. C'est pourquoi le concept de la « **station haptique** » est née. Qui n'est plus qu'une combinaison de types d'appareillages (« **Phantom** » et « **CyberGrasp** »)

La « **station haptique** » développée par IMMERSION™ est composée de plusieurs périphériques : « **DataGlove** », « **CyberGrasp** », et « **CyberForce** » qui est un exosquelette relié aux deux membres de l'utilisateur. Il acquiert le mouvement de l'utilisateur (place de poignet) et exerce les forces de réactions nécessaires lors de la manipulation d'objets virtuels. L'acquisition du mouvement est complétée avec le « **DataGlove** » qui tient compte de la position les différentes postures des mains. Les données récoltées serviront pour la construction des mains et des bras dans le monde virtuel permettant ainsi l'utilisateur de voir une représentation visuelle de lui par l'intermédiaire d'un avatar virtuel et facilitant ainsi l'interaction de lui avec les objets de son monde virtuel.

Un autre type d'appareil est utilisé aussi comme par exemple le « **DigiHaptic** » présenté par Casiez et al. [Casier2003], appareil « **haptique** » de trois degrés de liberté, ne visant pas à imiter l'anatomie de la main et dans les gestes mais seulement à gagner le temps dans interaction avec des objets rigides de la scène.

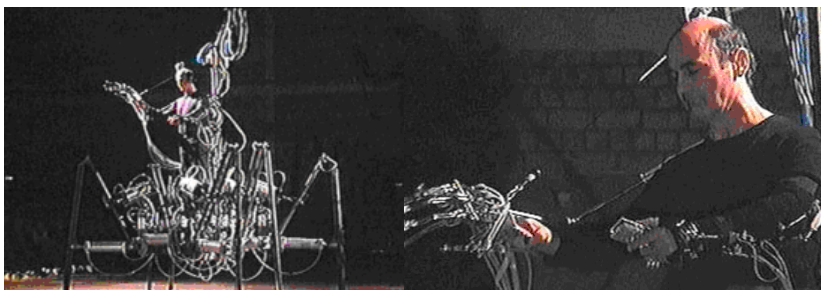


Figure 8 : Stelarc Exoskeleton 1999

La personne au centre de la structure, active la machine au moyen de bras munis de manipulateurs pneumatiques associés à des éléments mécaniques, électroniques. Cet exosquelette est commandé par geste.

<http://www.stelarc.va.com.au/exoskeleton/>. *L'objectif de marcher pour l'humain par l'intermédiaire de ce périphérique fait office de médiateur entre le désir de marcher par ce moyen de locomotion et l'opérateur.*

Hashimoto et Buss [Hashimoto1993] a aussi développé un système en s'inspirant des travaux rapportés dans la littérature par Howe [Howe1992] pour simulation de la force dynamique, en utilisant un gant de la réaction sensoriel avec 10 degrés de libertés (3 pour le poignet, 2 pour le pouce, 3 pour l'index et 2 pour le reste des doigts) qui se déplacent dans son ensemble. Un appareil semblable a été utilisé par Iwata [Iwata1992]. Les forces externes, tel que le poids d'un objet virtuel, sont appliquées sur la paume de la main de l'opérateur humain.

L'« *haptique* » comme on le sait fait aussi référence au sens du toucher [Fritz1999], une combinaison entre le tactile et force de réaction. La perception kinesthésique est une alternative permettant par exploration de ressentir les forces totales de contact. Comme c'est le cas dans l'application présenté par Springer et Ferrier [Springer1999]. Elle visait à rendre la sensation du touché lors de la manipulation d'objets réels à distance ou d'objets virtuels.

Les accessoires utilisés pour le tactile sont dans la majeure partie des cas pneumatiques [Burdea\_1\_1996][Stone1992], c'est à dire qu'à l'intérieur du gant sont placés des ballonnets gonflables. Lorsque l'utilisateur touche un objet, c'est-à-dire que, lorsque la réplique virtuelle du gant touche un objet virtuel, l'utilisateur ressent le contact de l'objet en question, sa dureté, sa rugosité, etc.

Nous pouvons aussi citer d'autres types d'accessoires : « vibrotactiles » ou « électrotactiles », avec plus ou moins de stimulation neuromusculaire.

Les systèmes « vibrotactiles » utilisent des vibrations situées au niveau des bouts des doigts, ce sont des sortes vibreurs électromagnétiques, des plaques de « micropins » [Burdea\_2\_1996]. Les systèmes « électrotactiles » utilisent des électrodes placées sur la peau pour stimuler localement nos récepteurs tactiles [Burdea\_1\_1996].

Une autre approche a été explorée, elle est basée sur l'usage de matériaux à mémoire de forme, c'est-à-dire que des petits fragments bien isolés d'un tel matériau peuvent se déformer de telle sorte qu'ils exercent une pression sur la peau.

La Réalité virtuelle combinée avec les technologies de l' « *haptique* », s'est révélée comme un outil excellent pour les applications de la « téléopération ». Des systèmes de « téléprésence » peuvent être prévus et élaborés afin de permettre à l'utilisateur d'opérer à distance et de ressentir comme s'il était physiquement présent sur le site éloigné. Les interfaces « *haptiques* » ont apporté, dans ce domaine, des améliorations indéniables.

La plupart des travaux en « *haptique* » consistent à créer des interfaces d'interactions efficaces en rendant des réactions « *haptique* » le plus proche de la réalité. Snibbe. [Snibbe2001] proposa plusieurs techniques de l' « *haptique* », c'est-à-dire destinés à fournir des moyens de contrôles, permettant de rehausser l'interface par des moyens numériques (vidéo, sonore), et reproduisant quelques-unes des propriétés physiques observées dans la réalité. La principale tendance actuelle pour rendre effectif des systèmes de « téléopération » est de regrouper plusieurs appareils « *haptiques* » et de les utiliser dans le domaine de la réalité virtuelle. Nitzsche al. [Nitzsche2001] dans leurs travaux ont essayé de pousser l'usage de l'« *haptique* » et de la réalité virtuelle à leurs limites. Ils proposèrent une interface pour les espaces de travail permettant d'interagir, sans restriction et réciproquement, avec des environnements éloignés.

### 2.4.3. Présentation de quelques périphériques

**Nous allons vous présenter maintenant quelques périphériques « haptiques ». (Voir figures ci-après)**

**Remarque :** Une vue d'ensemble plus détaillée à propos d'appareils de l'« *haptique* » tels qu'exosquelettes, gants, appareils portables, interfaces de locomotion et appareillages pour le corps entier, est à voir dans [Fisch2003].

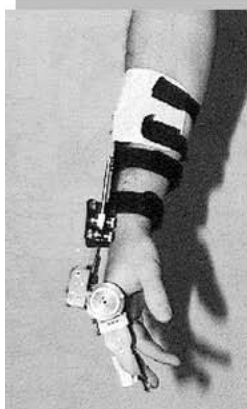
Utilisateurs et concepteurs de périphériques « *haptiques* »: CIE automobile (FIAT, ...), les militaires, PERCRO, Aerospace, AEC-NASA Technology Survey, Nothrop corporation, American Nuclear Society, VRSOLUTION, STROMFX, DaVinci, IntuitiveSurgical, ComputerMotion, IMMERSION, IRISA, Haption, SensAble, MPB Technologies, LIFL-L2EP, Sato-Koike, Reachin, FCS Control Systems BV, Force Dimension, Virtuouse 3D&6D, Laboratory for Intelligent Mechanical Systems (Mechanical Engineering Dept. 2145 Sheridan Road), MSL (Microdynamic Systems Laboratory),...



Virtuose 3D de la société Haption propose un retour d'effort sur les 6 degrés de liberté.



Le Rutgers Master I : Ils pèsent seulement 45g. L'action s'exerce grâce à des vérins pneumatiques. Les positions sont mesurées à l'aide d'un dataglove classique. L'échelle dynamique est excellente mais ces gants ne permettent cependant pas la totale fermeture du poing



Le Safire, proposé par Exos Inc., possède huit degrés de liberté (trois pour le pouce, trois pour l'index et deux pour le majeur). Il est basé sur des câbles



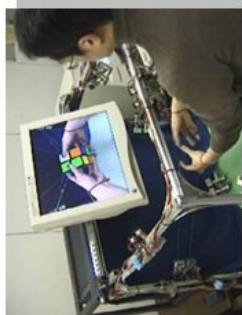
Exos Dextrous Hand Master, conçu au début des années 1990 au MIT est à l'origine un dataglove. On lui a ajouté des bobinages audio excités à 250Hz avec des amplitudes variables afin de fournir un retour tactile.



CyberTouch commercialisé par Virtual Technologies Corp. en 1995. Il pèse 144g. Il est basé sur six vibreurs électromagnétiques placés derrière les doigts et dans la paume. Ils produisent des vibrations de 0 à 125Hz pour une amplitude de 1,2N



Le HapticMaster de l'université de Tsukuba au Japon avec six degrés de liberté.

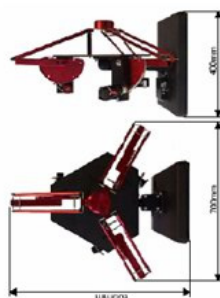


Le laboratoire Sato-Koike au Japon a développé un périphérique à câbles appelé SPIDAR. Chaque câble est motorisé et permet d'exercer un retour d'effort dans une direction.

Tactile Display proposé par le Forschungszentrum de Karlsruhe, il est basé sur le principe des afficheurs Braille. Forschungszentrum Karlsruhe



pincettes coelocopiques  
Immersion Corp.



La plate-forme delta est un périphérique 6 degrés de liberté basé sur une architecture de robot parallèle. Il est commercialisé par la société suisse Force Dimension sous deux versions à 3 et 6 degrés de liberté actifs.



Sarcos Dextrous Arm Master, à base d'effecteurs hydrauliques a été proposé par Sarcos INC., en collaboration de l'université d'Utah et le Naval Ocean Systems Center.



GRAB Haptic Interface



Force Feedback Exoskeleton  
Iowa State University



Sant'Anna School of University Studies and  
Doctoral Research PERCRO



Haptic Workstation (Immersion)  
[http://www.immersion.fr/HWS2\\_datasheet.pdf](http://www.immersion.fr/HWS2_datasheet.pdf)



## 2.5. DISCUSSION

Le but de la réalité virtuelle est de parvenir à une immersion totale afin d'ouvrir de nouveaux espaces de travail avec lesquels on pourrait entrer en adéquation tout aussi naturellement qu'avec la réalité. Pour cela les chercheurs déploient de multiples efforts dans l'illusion et l'utilisation de tous les canaux de communications de l'homme. Pour se faire une idée, on équipe l'utilisateur d'interfaces sensorielles afin de lui restituer les sensations nécessaires à son immersion et ainsi que d'autres dispositifs de communications. Plus le nombre de sens stimulés est grand (vision, ouïe, tactile), plus le sentiment d'immersion est de qualité, mais une bonne combinaison est nécessaire. Nous ne devons pas observer des contradictions au niveau de l'information perçue, même si notre cerveau a une grande capacité d'adaptation. Il s'agit donc là de nous convaincre que nous interagissons avec un environnement réel. Selon la classification de Morton Heilig, la vue est l'un des sens dont le pouvoir de monopolisation de l'attention est grand et de l'ordre de 70%. En effet, la vue est le canal sensoriel le plus important, car il permet de communiquer la plus grande quantité d'informations. La fréquence minimale pour détecter une continuité du signal vidéo est de 25Hz à 60Hz pour un confort visuel lors d'un mouvement de la tête. Et c'est par conséquent pour cela que les techniques de rendu 3D sont à la base des systèmes de rendu de la réalité virtuelle. Il va falloir l'utiliser sans négliger les autres sens comme par exemple celui du toucher.

La restitution des sensations tactiles peut se faire grâce à l'utilisation de gants spéciaux, permettant ainsi communiquer des informations sur les objets manipulés telles que leur forme, leur texture, leur vibration, leur mouvement, leur température. Dans le cadre du retour d'effort, 1kHz semble être une fréquence couramment admise pour obtenir une interaction stable en temps réel. Par conséquent, si le taux de rafraîchissement est trop faible, l'utilisateur pourra ressentir que quelque chose est anormal lors de la manipulation comme par exemple des tremblements des temps de latence. Même de la perte d'informations ponctuelles peut être perçue inconsciemment par l'utilisateur. Bien sûr, on peut naturellement utiliser une fréquence moindre, si la main n'est plus contrainte. Une des questions reste en suspens faut-il faire tourner l'environnement virtuel à la même fréquence ou la diminuer vers 25Hz ce qui risquerait d'altérer la perception « *haptique* ». Finalement, les interfaces « *haptiques* » requièrent une vitesse de mise à jour très élevée. Elles doivent fonctionner à des fréquences acceptables. Ce que nous pouvons constater dans le monde de la réalité virtuelle, c'est que le

principal problème en « *haptique* » est la détection de collision. Une gestion optimisée à son maximum des collisions entre les différents objets virtuels est donc nécessaire. Des algorithmes existants sont majoritairement destinés au graphisme (20-30 Hz) et ne peuvent gérer des scènes virtuelles aussi efficacement que les développeurs et les chercheurs l'auraient espéré (500-1000 Hz). L'efficacité couvre ici, robustesse, stabilité, vitesse et précision. Par ailleurs, un bon moteur physique est donc nécessaire si nous voulons effectuer des simulations physiques. De plus nous devons à chaque instant calculer toutes les forces de réactions et de répulsions en temps réel suivant le ou les champs définis dans notre monde virtuel. C'est-à-dire les propriétés de notre univers simulé.

Des questions qu'il faut se poser pour développer cette technologie « téléopératoire » Quelles mesures quantitatives peuvent être développées pour mesurer la qualité d'un tel système ? Comment peut-on avoir un système contrôle stable, de haute performance fonctionnant dans un environnement dynamique, ayant des délais raisonnables du point de vue de la communication, et qu'au niveau cinématique nous n'observons aucun de problème de singularité ? (Exemple d'un bras composé de deux parties, pour atteindre certains points de l'espace, il peut y avoir des solutions). Quelles sont les exigences pour que les interfaces de l'utilisateur soient efficaces quand il y a une grande différence dans la représentation de haut et bas niveau des connexions entre l'opérateur humain et le système contrôlé ? Que faire si le DOF du monde contrôlé ou esclave est supérieur au DOF du monde maître permettant de le contrôler ? La qualité du mode « téléopératoire » ainsi que celle de l'« *haptique* » dépendent surtout des avancées des conceptions techniques de part et autres du système.

### **2.5.1. Pourquoi les informations visuelles et tactiles sont importantes ?**

Pour sentir et percevoir la forme d'un objet, nous ne nous contentons pas d'un simple toucher. Nous devons le manipuler, pour pouvoir nous imprégner de sa forme, de sa nature, de la rugosité de sa surface, afin de nous construire une représentation mentale de cet objet. Il y a donc une forte corrélation entre manipulation et perception. [Srinivasan1997]. Ce sont ces mécanismes que nous devons comprendre pour que nous puissions, lors de la manipulation d'objet virtuel, provoquer chez l'utilisateur les mêmes schémas de pensée et de représentation mentale que ceux observés dans la réalité. D'autre part, il est nécessaire de bien comprendre les mécanismes de perception de l'information tactile afin de surmonter ces limitations. L'objectif est donc de manipuler un objet virtuel ou réel et que la représentation mentale de

cet objet faite par l'utilisateur soit la même que dans le réel. Cela nécessite une bonne modélisation combinée à des moyens « *haptiques* » et aussi une exploitation optimale de notre capacité visuelle à masquer plus ou moins les défauts.

### 2.5.2. Haptique contextuels passive ou active

Nous allons dire un petit mot sans entrer dans les détails sur l'utilisation de la combinaison de l'« *haptique passive* » et du visuel. Ce mode d'interaction est intéressant dans le cadre du concept du médiateur dont nous proposerons un peu plus loin, une conceptualisation utilisant cette technique.

Des expériences jusqu'à maintenant ont été menées afin d'explorer le rôle de l'« *haptique contextuels* » et des contraintes visuelles sur la manipulation de l'objets. Ces contraintes dans les environnements virtuels incluent les propriétés intrinsèques de l'objet manipulé [Wang1999]. Les contraintes contextuelles sont les informations l'environnement pour la manipulation de l'objet.

Il y a deux genres de contraintes de l'« *haptique contextuelles* » dans les environnements virtuels :

- Les contraintes de l'« *haptique actives* », provoquer par les appareils à retour de force et passives.
- Les contraintes de l'« *haptique passives* » sont rendues effectives avec de vrais objets dans des environnements augmentés. Par exemple, nous pouvons coupler comme ci-dessous une plaque réelle. La plaque réelle sert de support et d'aide à la sélection. Sur la plaque virtuelle nous pouvons ajouter des menus.

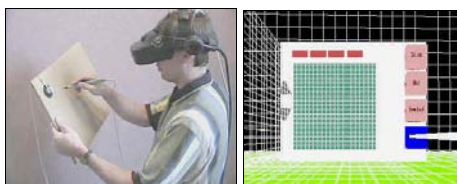


Figure 9 : Illustration du principe retour « *haptique passif* »  
Utilisation d'un stylo et d'une tablette comme moyen d'interaction.

L'utilisateur en immersion totale manipule la plaque virtuelle ainsi que la plaque réelle. L'utilisation de cette plaque permet d'exercer un retour de force passif sur l'utilisateur. Son étude a permis de voir les effets lors de l'interaction de l'« *haptique passive* » sur la manipulation d'objet de manière indirecte. Des recherches récentes en analysant les performances humaines dans l'interaction ont montré que l'« *haptique passive* » n'augmente pas seul le réalisme dans les environnements virtuels, mais aussi rehausse la performance humaine dans l'interaction [Linderman1999]. L'« *haptique passive* » augmente la rapidité de l'interaction, mais pas la précision. D'autre part les contraintes de l'« *haptique contextuelles* » sur la manipulation de l'objet sont essentiellement liées à un problème de degrés de liberté et aussi aux contraintes visuelles.

Finalement, pour les applications où la vitesse est une inquiétude majeure dans les simulations en temps réel, l'« *haptique passive* » est peut être une des solutions. Il faut donc creuser dans cette voie de recherche pour obtenir peut être de meilleures interactions. C'est pourquoi, les périphériques devraient être conçus pour simuler aussi bien l'« *haptique passive* » que l'« *haptique active* ».

### 2.5.3. Le pseudo-haptiques

Les périphériques « *haptiques* » seul ne suffisent généralement pas à satisfaire tous les besoins en interaction de l'utilisateur. L'interface « *haptique* » doit être accompagnée de multiples techniques, spécialement conçues pour restituer et exploiter les informations « *haptiques* ».

Le « *pseudo-haptique* » est peut être une solution, c'est un moyen supplémentaire pour faire croire à l'utilisateur dans la plupart des cas à une sensation de force. Des travaux dans ce domaine ont montré que certains dispositifs passifs combinés au rendu visuel peuvent restituer une information « *haptique* ». Comme par exemple un utilisateur était capable de distinguer des raideurs de différents ressorts virtuels avec uniquement comme moyen d'utilisation une simple combinaison de rendu visuel et d'une « *SpaceBall* ». L'utilisateur appuie sur une « *SpaceBall* » pour interagir avec des ressorts virtuels de raideurs différentes. L'expérimentation montre bien que la même force est utilisée sur ce périphérique alors qu'il pense appuyer plus au moins fortement sur le dispositif. Lécuyer a mis en exergue la notion

d'un rendu « *pseudo-haptique* » et « *d'illusion haptique* ». Des travaux de plusieurs chercheurs viennent conforter et compléter cette tendance. Ce qui ouvre la voie vers une autre façon de concevoir l'interfaçage « *haptique* », ne découlant pas forcément de la technologie de la téléopération. Peut-être qu'une combinaison des deux sera nécessaire pour obtenir une interaction réaliste. Mais il faut tout de même évaluer dans quelle mesure un « *retour haptique* » ou « *pseudo-haptique* » peut enrichir et améliorer les techniques d'interactions.

## 2.6. CONCLUSION

En guise de conclusion nous pouvons dire que les technologies de l'« *haptique* », en l'état actuel, sont loin d'être satisfaisantes. Nous distinguons trois parties fortement liées et étroitement interdépendantes qui peuvent être à l'origine des problèmes rencontrés entre l'utilisateur et son environnement informatisé : le système matériel de manipulation, le système informatique, le système de relation entre le système mécanique et le système informatique. Nous constatons que des problèmes techniques importants restent encore à résoudre. Cette technologie ne reproduit pas encore tous les paramètres permettant une immersion totale et naturelle. Penser que nous parviendrons à recréer le monde réel semble quelque peu utopique. Nous pouvons seulement l'approcher et utiliser des artifices (illusion, « *pseudo-haptique* ») pour que l'utilisateur y croit.

Ce chapitre nous a permis aussi de faire un état de l'art sur les interfaces, sur les interactions directes et indirectes dans un environnement virtuel. Grâce à cela nous pouvons peu à peu déterminer nos choix, ainsi que des solutions. La flexibilité, l'adaptabilité (c'est-à-dire donner les outils nécessaires pour que l'utilisateur puisse modifier l'interface), « l'adaptativité » (propriété désignant la capacité à se modifier sans intervention explicite), la « *généricité* », « l'affordance » (propriétés actionnables entre le monde et un utilisateur) sont des éléments clés et indispensable à intégrer dans notre concept du médiateur.



# CHAPITRE 2 : PROJET VIRTUAL

L'objectif de ce chapitre est de montrer par le biais d'une application concrète, c'est à dire un simulateur de voiture développé lors d'un projet Européen dénommé VIRTUAL, les points délicats auxquels nous avons été confronté : les différentes interfaces, le degré d'immersion, l'interaction réelle et virtuelle, l'« *haptique* », les problèmes relatifs à la détection de collisions, les problèmes de synchronisation de l'application au niveau réseau ainsi que l'implémentation d'un moteur dynamique pour la voiture. Les résultats ainsi obtenus, nous ont donné des éléments non négligeables pour la conceptualisation et l'élaboration de notre interface médiatrice.

Rappelons que l'idée du médiateur dans le cas de notre simulateur se présente sous la forme de périphériques hardware réels ou d'un avatar humain servant d'instrument médiateur entre le conducteur et la voiture.

## 3.1. INTRODUCTION

Le projet VIRTUAL était un projet réunissant plusieurs partenaires Européens dont l'université de Genève MIRAlab, PERCRO de Pise, le centre de recherche FIAT en Italie, Fraunhofer- Institute FhG, Loughborough University, Darmstadt university (IAD), et le VRlab-EPFL.

Jusqu'à maintenant l'expérimentation ergonomique des véhicules est traditionnellement effectuée sur des prototypes conceptuels différents. Cette approche impose de sévères contraintes en termes de temps et coûts qui limitent le nombre de tests que l'on peut faire. Divers environnements ont donc été créés et évalués en réalité virtuelle comme

solutions alternatives au prototypage de véhicules. La réalité virtuelle offre la capacité à réaliser très facilement des solutions conceptuelles différentes.

L'objectif principal du projet était par conséquent de développer une plate-forme automatisée, basé sur la réalité virtuelle, afin de pouvoir pour tester l'ergonomie des véhicules et enfin de développer des critères pertinents permettant d'estimer la qualité du véhicule (visibilité interne et externe, manœuvrabilité, accessibilité des moyens de contrôle et ...). Grâce à des séquences de contrôles prédéfinis, un utilisateur allait pouvoir tester différents points d'ergonomie, tels que la visibilité des compteurs, la maniabilité et l'accessibilité des contrôles, ainsi que la facilité de conduite durant des manœuvres types comme la marche arrière ou le parage. Par exemple, une séquence de tests consistait à conduire sur une route en évitant le chien qui traverse, à franchir un carrefour, à parquer sa voiture entre deux véhicules ou dans un garage, etc... Ces tests sont donc simples, mais grâce aux résultats obtenus, les ingénieurs pourront simplifier, sécuriser et améliorer la conduite des nouveaux véhicules.

Ce projet a donc été développé en 3 phases. Trois simulateurs de conduite, présentant différents niveaux d'immersion (« non-immersif », « semi-immersif » et enfin totalement immersif) ont été développés avec différents équipements hardware. *Nous avons pu grâce à cela tester plusieurs types d'interfaces et mesurer ainsi leurs efficacités.*

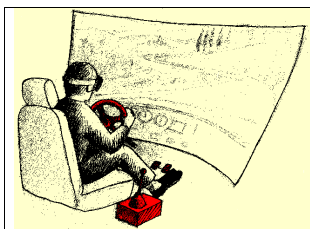
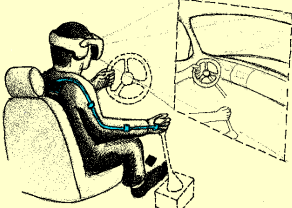
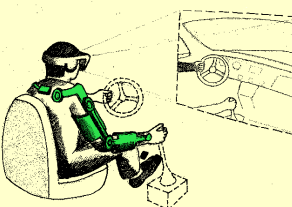


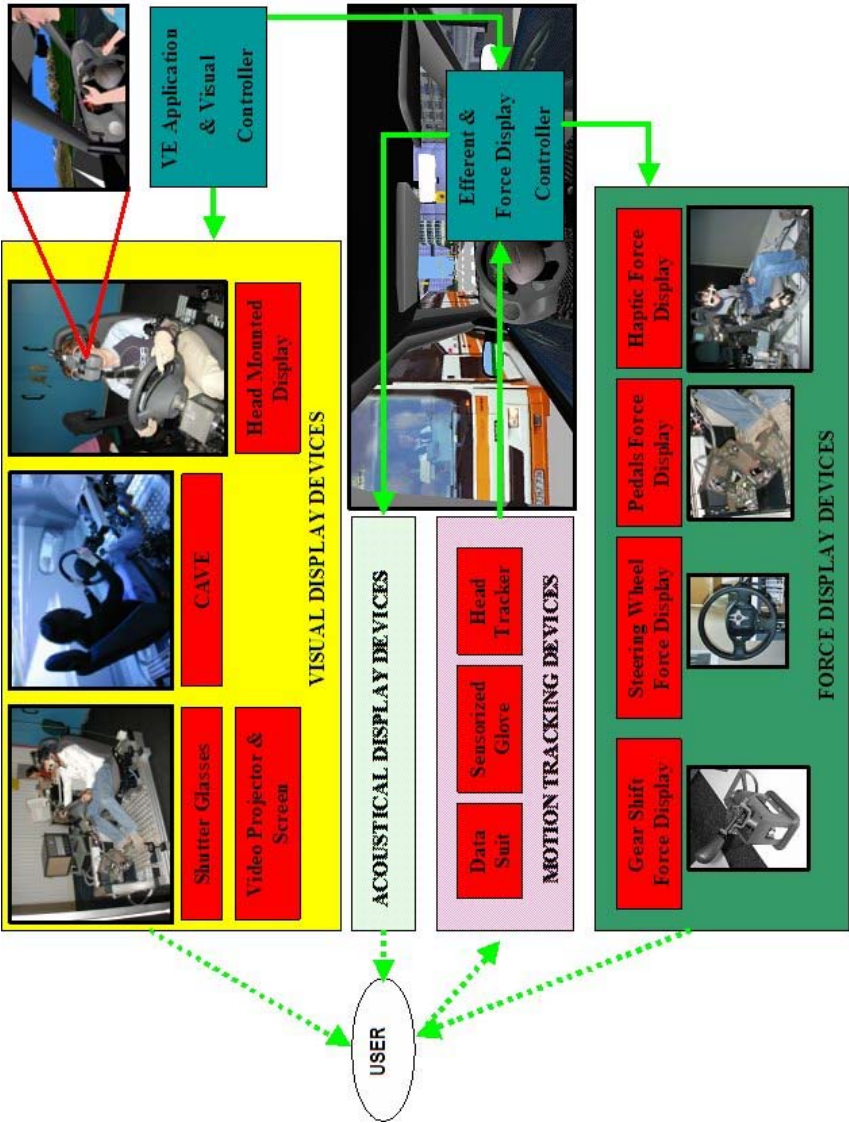
Figure 10 : Le système 1

Le système 1 utilise un grand écran stéréo et des commandes « *haptiques* ». Il a été utilisé pour tester la visibilité intérieure et extérieure, ainsi que pour des tests de conduite simplifiés. L'environnement matériel utilisé : un projecteur, des lunettes stéréo pour rendre la troisième dimension, des périphériques à retour d'effort utilisés pour la conduite.



	<p>Le système 2 est basé sur le "feedback" visuel employant une simulation immersive. Il visait à introduire une interaction tactile virtuelle en donnant une représentation visuelle des mouvements du conducteur à l'aide d'un avatar humain. Pour cela des équipements et techniques appropriées ont été utilisés (traqueur, interprétation du mouvement, casque de réalité virtuelle (<b>Head Mounted Display</b>))</p>
	<p>Le troisième système utilise une interface « <i>haptique</i> » qui remplace les commandes « <i>haptiques</i> » de la maquette. Au final, nous sommes arrivés à un système utilisant un HMD ou bien un système CAVE pour afficher l'environnement et tous les équipements de contrôle ont été créés virtuellement, y compris le volant. Ce qui nous a permis de voir les réactions d'interaction directe avec ou sans retour de force des utilisateurs face à des objets virtuels. Pour le retour d'efforts nous avons utilisé un exosquelette spécialement fabriqué et développé par PERCRO.</p>

Nous verrons un peu plus loin la description des différents systèmes, les résultats obtenus ainsi que les tests qui ont été fait avec des candidats novices ou experts en réalité virtuelle. Ci-après, nous vous présentons une synthèse globale des différents systèmes.



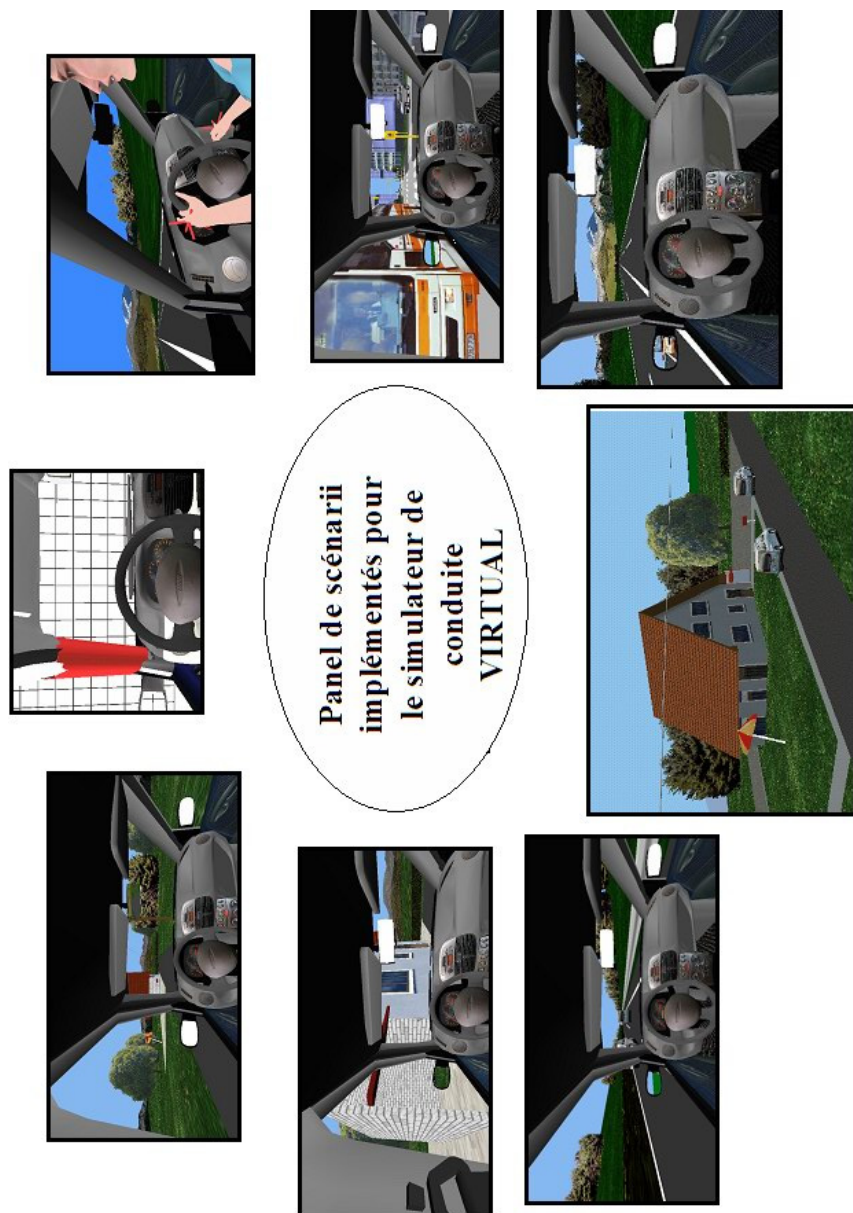


Figure 13 : architecture générale du système.

Chacun des partenaires avait un rôle bien défini, le VRlab-EPFL devait développer les différents modules du logiciel et y intégrer les périphériques matériels développés à PERCRO, ainsi que les humanoïdes de MIRAlab (Genève). L'environnement comprenait donc les modèles 3D utilisés, ainsi que tous les outils nécessaires à l'animation de l'environnement.

Des tests ont été réalisés tout au long du projet. Des candidates avec des degrés d'expériences différentes au niveau de la conduite ont été évaluées dans des tâches différentes de pilotage. Les performances objectives et évaluations subjectives de l'ergonomie du véhicule et du simulateur ont été comparées et on permit de le valider [Ruspa\_1\_2002], [Ruspa\_2\_2002]. Beaucoup de tests ont été réalisés, tant au niveau de l'analyse des effets de nausées le fameux « *cybersickness* » (terme proposé par McCauley et Sharkey), de la qualité de la simulation et l'évaluation de la qualité du système VR (présence spatiale, qualité de l'interface de l'utilisateur et émotivité). Pour cela, un instrument d'évaluation a donc été développé pour ce projet afin d'analyser et de recueillir les données. La validation de notre système de conduite était une partie du coeur du projet dont nous pouvons voir certains résultats dans le document de Scheuchenpflug [Scheuchenpflug2002].

Pour tous les trois systèmes développés, beaucoup de paramètres ont été recueillis, et mesurés par des méthodes appropriées. C'est-à-dire des données qui décrivent le comportement du simulateur de voiture, les aspects de l'ergonomie (visibilité interne et externe, portée et usage des contrôles), l'évaluation subjective d'acceptation du système (qualité des tâches exécutées, « *cybersickness* » etc..), et enfin la notion de présence dans le simulateur de conduite [Krauss2001], [Ruspa\_3\_2002].

Finalement, ce projet européen a permis de tester la faisabilité des différents systèmes, ainsi que de l'utilité de la réalité virtuelle dans la création de nouveaux véhicules.

### 3.2. PRESENTATION DES DIFFERENTS SYSTEMES

Nous avons développé donc trois différents systèmes. Nous vous présentons maintenant les différents systèmes.

#### Le Système 1 : Non Immersif



Figure 14 : « *Mock-up* » physiques, périphérique de contrôle.

Ce système est composé :

- De périphériques de contrôle : un levier de changement de vitesse, un volant, les pédales (frein, accélérateur, débrayage) avec retour d'efforts [Angerilli2001], [Frisoli2001].
- D'un système pour traquer les mouvements de l'utilisateur : InterSence Tracker [InterSence] dispositif utilisé pour savoir dans quelle direction était porté le regard de l'utilisateur pour activer ou non un des trois miroirs.
- D'un système de visualisation : un projecteur CRT (BARCO 909 modèle : 120 Hz) en mode stéréo avec utilisation de CrystalEyes 3 CE-3.

#### Remarques :

La carte graphique était un élément clé conditionnant la complexité des algorithmes (animation des véhicules autonomes, les comportements du véhicule propre : le moteur, la gestion des collisions, et de l'interaction) dans ce système. Le port de lunettes pour la stéréovision, permettait à l'utilisateur de percevoir l'environnement artificiel en 3D. Les

défauts de système se situaient surtout au niveau de la vision car le compteur de vitesse de la voiture virtuelle était à peine lisible.

Concernant la capture du mouvement de l'utilisateur, nous avons du utiliser un traqueur à ultrason car il était impossible d'utiliser un traqueur magnétique de type Polhemus à cause des interférences dû aux maquettes physiques (« *Mock-up* »). En effet, la présence du métal perturbe les mesures de positionnement et d'orientation de la tête de l'utilisateur. Cependant les sondes ultrasoniques ont d'autres problèmes liés à la sensibilité à certains sons ou à la liaison avec le récepteur, mais ce ne sont que de petites fluctuations acceptables car en définitive, cela permet tout de même de faire une manipulation un peu plus précise.

### Le système 2 : Immersif



Figure 15 : Configuration « HMD » ;

L'avatar contrôlé par le conducteur en milieu immersif.



Ce système est composé :

- Les périphériques de contrôle : levier de changement de vitesse, volant, pédales (frein, accélérateur, débrayage) avec retour d'effort en option.

- D'un système pour suivre les mouvements de l'utilisateur (InterSense Tracker) : traqueur permettant de capturer les positions et l'orientation de la tête afin de pouvoir calculer ce que regarde la personne dans la scène virtuelle en stéréovision. Puis, un autre traqueur "**DataSuit**" capturant comme le principe du "**DataGlove**", le mouvement des membres supérieurs de la personne conduisant la voiture virtuelle. Et enfin deux "**DataGlove**" pour obtenir l'orientation des différents doigts.
- D'un système de visualisation : casque de réalité virtuel (HMD).
- De 2 PC : INPUT-PC et OUTPUT-PC. OUTPUT-PC gère tout ce qui est la partie visuelle et affichage de l'environnement virtuel. Le INPUT-PC, c'est le PC maître qui contrôle tout le système ; il reçoit et gère les paramètres des différents périphériques hardware de contrôle. La connexion entre les deux machines se fait par TCP/IP



Figure 16 : Le « **DataSuit** »

**Remarques :** La personne immergée dans le monde virtuel est représentée par un avatar (un humain virtuel). Quand l'utilisateur conduit la voiture, ses mains réelles sont représentées par des mains virtuelles. Les mouvements de ses mains, de ses bras sont calculés à partir des informations fournies par le « **DataSuit** » ou par un autre dispositif : traqueur positionné sur les mains fournissant position et l'orientation. Ces paramètres permettent de calculer par cinématique inverse la posture des membres supérieurs de l'avatar. Le « **DataSuit** » est un périphérique permettant en temps réel de mesurer les angles entre les membres adjacents des bras et mains.



### Le système 3 : Totalemment immersif

#### Pour VIRTUAL système 3

- CAVE avec le périphérique de contrôle de base ou Exosquelette.
- HMD avec Exosquelette

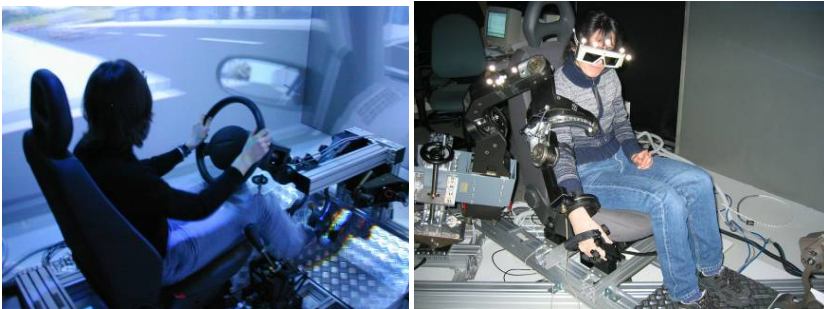


Figure 17 : Les différents systèmes

Système « *Mock-up* » avec « *Shutter glasses* » à droite ; Système CAVE avec « exosquelette » à gauche ; En bas des photos du système avec l'exosquelette.



Ce système est composé :

- De périphériques de contrôle : les pédales (frein, accélérateur, débrayage) et exosquelette avec retour d'effort [Frisoli2005] venant se mettre autour du membre supérieur de l'utilisateur.
- D'un système pour suivre les mouvements de l'utilisateur : (Infrared Tracking System : IR-optical Tracking System ARTtrack) ou (Magnetic Tracking System : MotionStar Wireless) traqueur permettant de capturer les positions et l'orientation de



la tête afin de pouvoir calculer ce que regarde la personne dans la scène virtuelle. Puis, un autre traqueur pour calibrer le système utilisant l'exosquelette.

- D'un système de visualisation : casque de réalité virtuelle (HMD) or système CAVE.
- De 4 PC et d'un SGI ;
  - PC1 est utilisé pour la gestion des collisions, des pédales, et de l'exosquelette ;
  - PC2 est utilisé pour gérer le fond sonore ;
  - PC3 est utilisé pour le système de traqueurs optiques installés dans le système CAVE ;
  - PC4, le PC maître, est utilisé pour gérer, synchroniser tout le système, pour exécuter les scénarios, pour enregistrer tous les paramètres analysant le comportement du conducteur ;
  - le SGI est utilisé pour le rendu de la scène sur tous les murs de la CAVE (Pour la configuration HMD le SGI est remplacé par un PC).

### Remarques :

Dans le système utilisant un exosquelette avec retour d'effort, la personne immergée dans le monde virtuel agit sur l'environnement. Elle peut interagir avec les périphériques de contrôles de la voiture : toucher, saisir et tourner le volant virtuel, changer les vitesses.

Le système CAVE (PCs et SGI Onyx 1000) est un système de projection en mode stéréovision composé de miroirs de projecteurs, projetant sur un système de 4 écrans l'environnement virtuel. Ce système permet de donner l'illusion d'être debout dans un monde virtuel. On utilise un système de traqueurs optiques infrarouges pour suivre les mouvements de l'utilisateur. Pour acquérir un effet 3D, l'utilisateur doit porter des lunettes de stéréovision. Nous avons donc développé un module qui s'occupe de calculer correctement les images stéréo. Les problèmes étaient de faire coïncider l'image du volant virtuel avec le volant réel, et de régler les problèmes de superposition des images stéréo dans les coins de la CAVE.

Une sonorisation 3D a été aussi installée pour augmenter le degré d'immersion. Il inclut une "Peavey Media Matrix MM8840» avec 8 in/output channels. Le son fournit un bruit réaliste du moteur de voiture.

Un système de traqueurs optiques infrarouges «*ARTirack&Dtrack* » analyse avec précision les moindres mouvements de l'utilisateur.

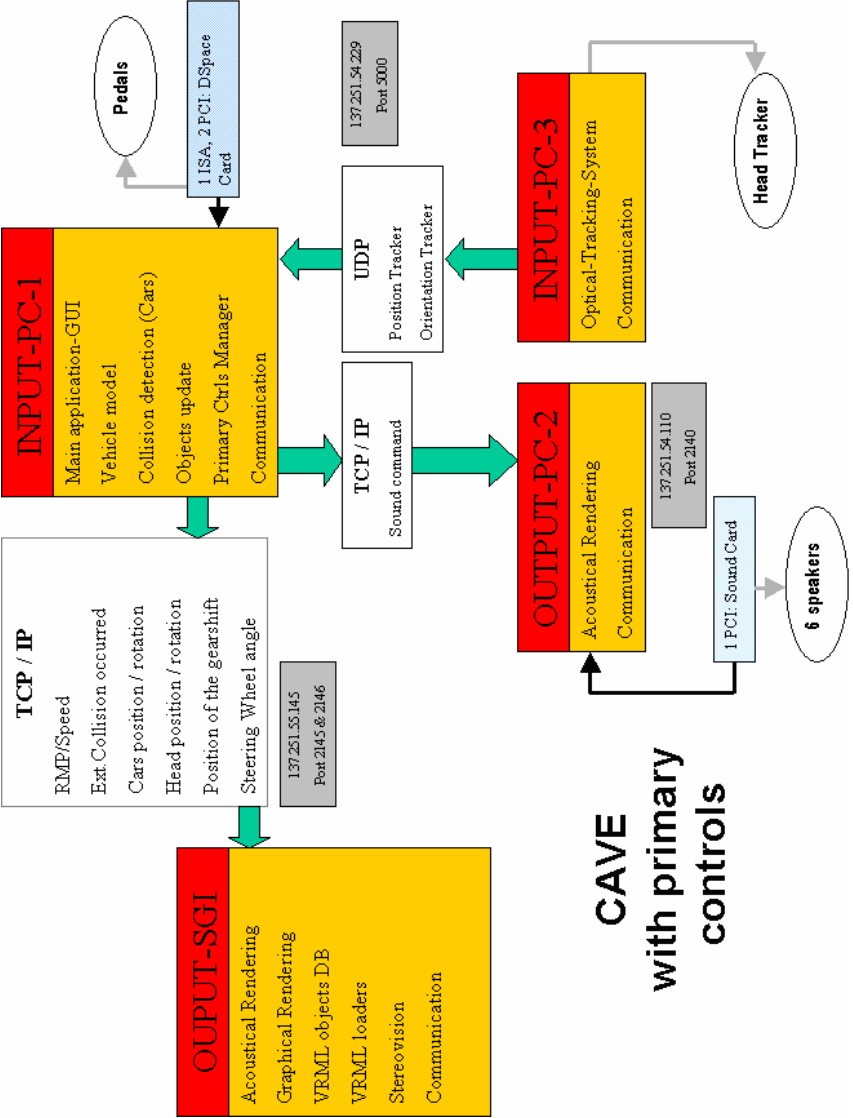
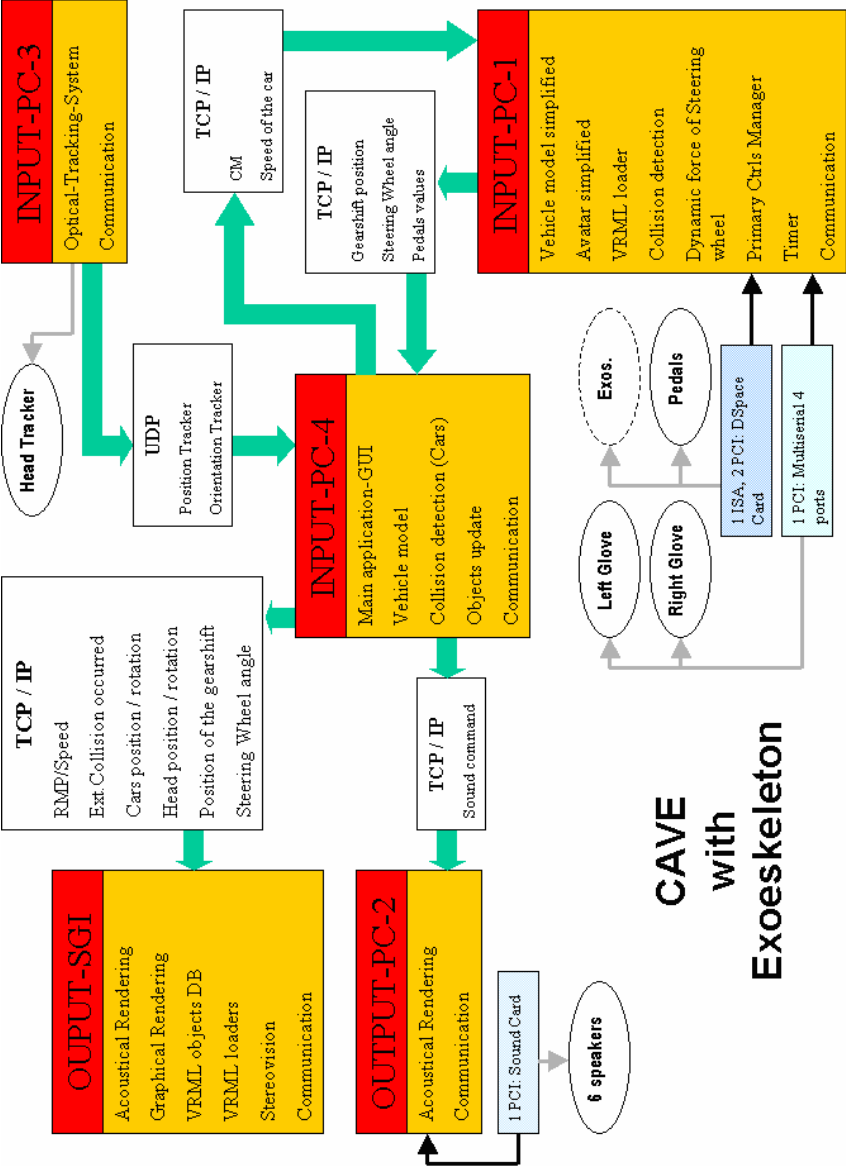


Figure 18 : Organigramme des différents systèmes



### 3.3. PRESENTATION DES MODULES COMPOSANT LES SYSTEMES

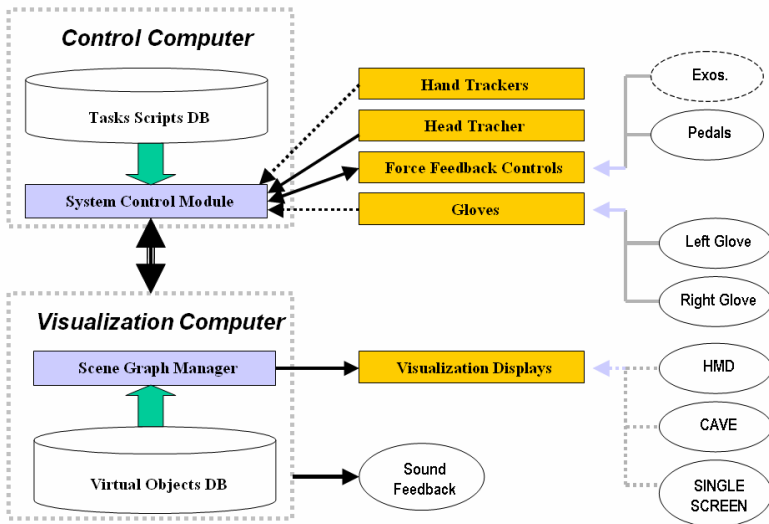


Figure 19 : Diagramme simplifié

Nous avons choisi de développer un système capable de simuler des tâches spécifiques, donc configurables à partir de scripts, interprétés lors de l'exécution. Ces scripts respectent la syntaxe Python, mais font aussi appel à des méthodes spécialement définies dans l'application. L'idée principale est que la configuration ainsi que la gestion du système se fait de l'extérieur. Pour chaque tâche, deux scénarios doivent être donnés :

- **un scénario d'initialisation** qui configure les paramètres du système c'est à dire les paramètres d'initialisation de la caméra, les paramètres d'initialisation des périphériques hardware, ainsi que le chargement des éléments de la scène et des paramètres du scénario.
- **un scénario joué en temps réel** qui contrôle le comportement de tous les objets animés dans la scène pendant la simulation. Les comportements des voitures autonomes sont contrôlés à partir du scénario.

En effet, les fonctions disponibles dans Python permettent de développer facilement pour notre environnement des scénarios liés à notre système. Plusieurs scripts ont été mis au point pour diverses situations par exemple : S'arrêter à feu tricolore, tourner à gauche, manœuvrer dans un espace restreint : garage, parking, conduire sur l'autoroute, s'engager sur un carrefour. Le comportement global des véhicules résulte d'une combinaison de plusieurs comportements de base (trajectoire à suivre, action d'évitement de l'obstacle, etc.) définis des scripts python. Nous avons ainsi établi des règles de conduite.

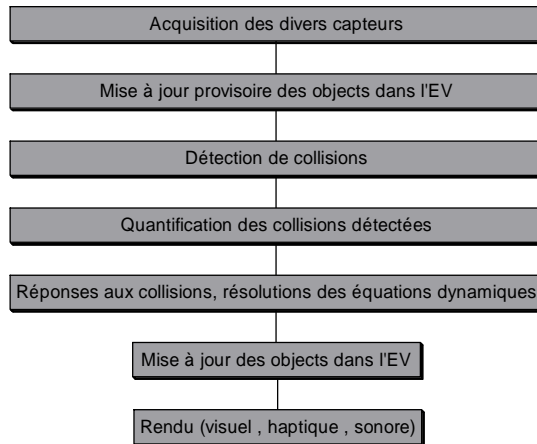
Maintenant nous allons vous présenter les différents modules et sous modules qui composent notre application.

### 3.3.1. Le module du contrôle du système

Ce module est composé :

- D'une interface de sélection de script permettant de sélectionner le scénario désiré. Les scénarios sont chargés à partir d'une base de données locale qui contient tout les données nécessaires. .
- D'un module qui s'occupe de la partie hardware mettant à jour les paramètres du véhicule virtuel, c'est à dire des paramètres du volant, pédales, levier de changement de vitesse. Il s'occupe aussi de lire les paramètres des « **DataGlove** », de l'exosquelette. Ces valeurs sont utilisées pour animer l'avatar (dans le cas où un HMD est utilisé). À chacun instant, ce module s'occupe de coordonner les mises à jour tant au niveau hardware que de la gestion du décor. Dans le cas où le simulateur utiliserait plusieurs machines, l'ordinateur de contrôle s'occupe d'envoyer les directives de manière synchronisée aux autres machines afin qu'elles puissent afficher correctement les éléments du décor.
- Un module qui s'occupe de calculer les forces au niveau des périphériques de contrôles. **Exemple** : le volant devient de plus en plus difficile à tourner quand la vitesse augmente.

Ce module s'occupe de la synchronisation et de l'accomplissement de toutes ces étapes présentées ci-dessous :



Dès qu'une étape est terminée, on retourne à l'étape acquisition des divers capteurs et ainsi de suite.

### 3.3.2. Le module graphique de la scène et de visualisation

Ce module s'occupe de charger les décors, de calculer la scène en mode stéréo suivant le mode configuration visualisation HMD et système CAVE. Le calcul de la scène se fait suivant les valeurs provenant d'un traqueur posé sur la tête du conducteur (position et orientation). Ces valeurs ne servent pas seulement pour afficher et construire le décor qu'il observe, mais aussi d'indicateur permettant d'analyser le comportement de l'individu lors de la conduite.

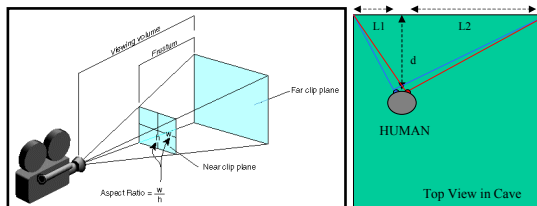


Figure 20 : Comment fonctionne la stéréovision ?

Pour chacun des écrans, nous calculons les valeurs suivantes pour la camera virtuel : SetNearFar(d,distance, clip far), le frustum caméra makePersp (left, right, bot, top) avec left=-L1 et right=L2. Puis on place la caméra virtuelle dans le véhicule calculé au niveau des yeux de la personne. D'autres calculs et corrections sont nécessaires pour obtenir une bonne stéréo à l'intérieur du système CAVE. Il fallait éviter tout chevauchement ou trou qui perturberait la qualité de la simulation (voir figure ci-dessous). Notre cerveau doit donc pouvoir reconstruire sans effort les objets 3D, surtout au niveau des coins et des bords des écrans. Pour résoudre ce problème, nous avons trouvé cette solution.

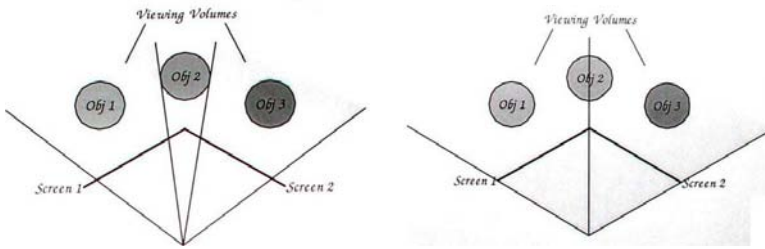


Figure 21 : Visualisation des problèmes de projection sur les murs

Formulation sommaire des paramètres de correction pour observer une bonne stéréovision pour le système CAVE :

Soit les paramètres suivants : V : position de la tête de l'utilisateur ; R et L la position de yeux ; Q : représentant le quaternion c'est à dire contient l'orientation de la tête de l'utilisateur ; IOD : représente la distance intraoculaire qui est de l'ordre de 5 à 7 cm suivant les adultes.

$$R = V + \Delta; L = V + \Delta;$$

$$\Delta_x = \frac{IOD}{|Q|} (q_x^2 + q_z^2)^{1/2} \sin(q_w)$$

$$\Delta_y = \frac{IOD}{|Q| (q_y^2 + q_z^2)^{1/2}} (q_z |Q| \cos(q_w) - q_x q_y \sin(q_w))$$

$$\Delta_z = \frac{IOD}{|Q| (q_y^2 + q_z^2)^{1/2}} (-q_y |Q| \cos(q_w) - q_x q_y \sin(q_w))$$

$$|Q| = (q_x^2 + q_y^2 + q_z^2)^{1/2}$$

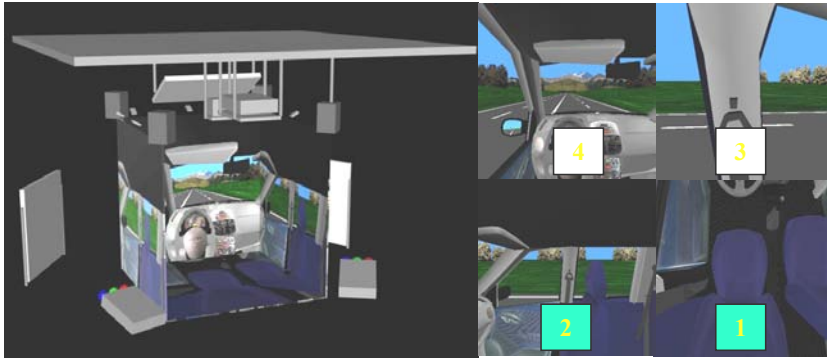


Figure 22 : Schématisation du système CAVE

Les images de chaque panneau écran sont calculées simultanément La synchronisation est pilotée par le PC maître.

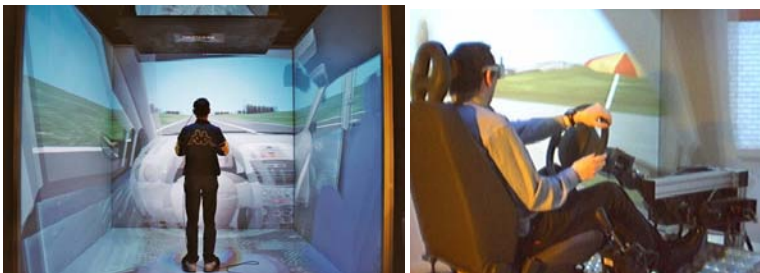


Figure 23 : Le système CAVE.

**Remarque concernant la partie graphique :** Tout au long de l'implémentation du simulateur de conduite, nous avons porté nos efforts sur un compromis entre le nombre de polygones et le réalisme du comportement des objets dans la scène. Des efforts ont été accomplis sur les différents modèles de l'habitacle du véhicule, grâce à une combinaison subtile de formes polygonales et de textures.



### **3.3.3. Le module modèle physique pour la voiture**

Ce module s'occupe de tous ce qui est relatif aux mouvements des voitures ainsi que le moteur de notre voiture. Nous avons utilisé et élaboré un modèle physique simplifié, basé sur un modèle dynamique pour l'animation du mouvement des véhicules. Puis à partir des divers paramètres concernant la voiture, nous avons construit notre moteur physique simplifié qui épouse à peu près le même comportement que dans le réel. Il faut noter que la création d'un moteur physique réaliste aurait pris beaucoup de temps et que ce n'était pas notre objectif. C'est pourquoi nous avons implémenté un moteur très simplifié. Nous n'entrerons pas dans les détails sur les formules physiques et mécaniques utilisées car ceci ferait l'objet d'un grand nombre de pages sans intérêt relatif à l'idée développée dans ce sujet. Nous vous invitons pour plus d'information à consulter un ouvrage sur la modélisation du comportement physique d'une voiture.

### **3.3.4. Le module de gestion de collision**

Ce module est un élément important dans notre simulateur, car il permet d'une part de d'éviter les collisions entre les différentes voitures, piétons et objets de la scène et d'autre part de donner à l'utilisateur via l'intermédiaire d'un avatar humain de manipuler les périphériques virtuels internes de conduite de la voiture. Cela nous a conduit tout naturellement à calculer aussi des collisions entre l'avatar humain et les périphériques virtuels de contrôle (volant, levier de changement de vitesse)

Afin d'optimiser au maximum le système et avoir une fluidité dans l'exécution, nous devons déterminer la meilleure approche pour gérer ces problèmes de détection de collision. La plupart des scènes de la réalité virtuelle comportent beaucoup de facettes, ce qui implique que si nous devons effectuer une manipulation en temps réel, le moteur de collision devait être très puissant. Par conséquent des choix, et sacrifices ont du être faits au niveau de la précision.

La détection de collisions est aujourd'hui un thème de recherche majeur en informatique, car elle est une des étapes critiques dans un système de simulation dynamique et nécessite un temps de calcul très important par rapport à celui nécessaire au calcul du mouvement et des déformations d'objets.

L'exigence de la détection en temps réel impose des algorithmes performants et peu coûteux en temps machine. Elle doit être la plus fine possible afin d'éviter visuellement l'interpénétration des objets. Il faut noter que malgré de constantes améliorations même au niveau « hardware » les algorithmes de détection de collision représentent toujours un goulot d'étranglement dans différents domaines de simulation dynamique. Déterminer les interactions entre  $n$  objets est un problème dont la complexité est en  $O(n^2)$ . La plupart des algorithmes de détection de collision existant se bornent aux objets rigides et sont rarement adaptés aux objets déformables.

De nombreux algorithmes de détection de collision se bornent aux objets convexes. Mais dans le monde qui nous entoure, il n'y a pas que des objets convexes. Une des solutions consiste à décomposer l'objet concave en un ensemble d'objets convexes. Le problème c'est que cela augmente les tests et qu'il n'est pas possible d'utiliser cette méthode pour les objets déformables.

De plus, si on l'applique à une simulation physique, il est nécessaire de simuler précisément les différents phénomènes physiques afin d'améliorer le réalisme du monde virtuel. Pour rendre une simulation réaliste, la gestion de la réaction impose une finesse de détection. Il est donc nécessaire de déterminer le plus précisément possible les éventuelles zones de l'objet en collision.

**Nous avons du répondre à de nombreuses questions :**

- ✓ *Comment détecter la collision entre les objets de la scène ?*
- ✓ *Comment localiser les éléments du contact ?*
- ✓ *Comment calculer les forces résultantes de ce contact ?*

Mais au préalable, nous nous sommes demandés s'il fallait calculer les collisions objets par rapport à des facettes ou par une modélisation par encapsulation de l'objet à l'aide de primitives géométriques. D'où, ce nouveau questionnement :

- ✓ Comment faire une détection de collision de fréquence élevée et précise entre la main et l'objet ?
- ✓ Comment manipuler l'objet ?
- ✓ Comment saisir l'objet ? Faut-il une grande force de contact pour modifier son état
- ✓ Comment corriger la posture de la main ?
- ✓ Comment calculer les forces au niveau de la main (nécessaire si on utilise un périphérique avec retour de forces).

Après avoir fait un état de l'art sur les moteurs de collisions disponible sur le marcher, comme par exemple : V-COLLIDE, RAPID, PQP, SOLID, QuickCD, V-Clip, SWIFT et dégagé les multiples possibilités concernant les méthodes de collision, nous avons décidé d'implémenter deux méthodes. *(A noté que des études ont été aussi faites pour comparer et classifier les différents algorithmes GJK, Lin-Canny... de détection de collision et librairies disponibles sur le marché. [Mirtich1998] , [Larsen2000] , [Reggiani2002].)*

Nous avons utilisé V-COLLIDE [V-COLLIDE], [Hudson1997] d'une part pour détecter les collisions entre les voitures de la scène. Sachant que V-COLLIDE est une librairie de détection de collision pour les environnements virtuels vastes. Elle renvoie uniquement le statut des contacts (et pas de distance). De plus elle utilise un arbre OBB et des tests exacts pour déterminer si une paire de triangles se chevauche ou non. Le processus mis en œuvre tient compte de la cohérence temporelle et se révèle par conséquent efficace pour les animations et simulations de mouvements. L'utilisation de ce moteur de collision, nous a permis de gérer un grand nombre d'objets mobiles dans la scène avec peu de triangles. Ce qui nous a permis de gagner en fluidité, sachant qu'une précision fine n'était pas requise et qu'il fallait plutôt focaliser les ressources disponibles sur la détection de collision entre l'avatar humain et les objets internes à la voiture. La détection devait être précise et rapide avec le moins calculs possibles. C'est pourquoi nous avons préféré une méthode utilisant des fonctions implicites et primitives de bases (parallélépipède, « métaformes », et autres objets pouvant être décrit sous forme d'équations) pour détecter les collisions lors de l'interaction entre l'homme et son véhicule virtuel. Nous pensons qu'englober un objet « facetisé » mais

lisse dans la réalité, en une combinaison de primitives de bases semble être aussi une bonne approche. C'est pourquoi, nous avons implémenté cette idée dans notre système.

### 3.3.5. Le module de collision voiture et objets de la scène

Une scène est constituée d'objets mobiles (voitures et piétons). Qui dit mobilité, dit susceptible d'entrer en collision. Par conséquent, il va falloir trouver un bon moteur de gestion de collisions, rapide et efficace, ne prenant pas beaucoup de ressources au système.

L'algorithme de détection de collisions utilisé pour la gestion des collisions voitures et de piéton est la librairie V-COLLIDE développée en 1997 [Hudson1997] à l'université de Caroline du Nord (UNC) à Chapel Hill, USA. V-COLLIDE est essentiellement composée de deux couches distinctes : une détection de contact de paires d'agglomérats de triangles, et d'une détection de contact de triangles. Cette seconde partie est en réalité la librairie RAPID (Robust and Accurate Polygon Interference Detection) développée également à l'UNC. Dans un premier temps, V-COLLIDE crée une structure hiérarchique, un arbre de « bounding boxes » (OBB tree), selon le schéma suivant :

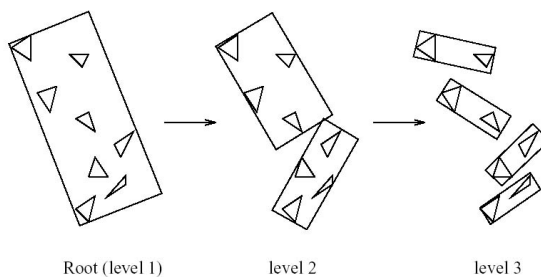


Figure 24 : Regroupement par paires de triangles

Chaque agglomérat de triangles est divisé en deux, selon les distances entre les triangles, jusqu'à n'obtenir que des paires de triangles. Lorsque par la suite le programme interroge V-COLLIDE pour savoir si des collisions ont eu lieu, l'arbre est parcouru récursivement de la racine aux feuilles. L'avantage d'utiliser des arbres de « **bounding boxes** » devient assez évident, puisque deux objets qui ne se touchent pas du tout vont être éliminés d'office par le premier test sur les racines des arbres de chacun des objets. V-

COLLIDE utilise encore une optimisation supplémentaire, puisque la routine, qui teste les collisions entre les agglomérats, mémorise les collisions qui ont eu lieu dans un certain intervalle dans le passé pour les tester à nouveau avec une priorité plus grande.

### 3.3.6. Le module gérant l'avatar humain et ses collisions

Ce module calcule à chaque instant les collisions entre l'avatar humain et les objets internes de la voiture. L'avatar humain est un outil permettant au conducteur d'interagir avec la voiture virtuelle. Pour la configuration CAVE, l'Avatar humain n'est pas affiché, car il serait une gêne pour l'utilisateur, mais il est tout de même utilisé sans être affiché pour détecter des collisions avec les contrôles virtuels. Dans le cas de la configuration HMD, nous affichons l'avatar. L'avatar subit des corrections nécessaires avant qu'il soit affiché. L'idée est de garder toujours une cohésion dans la visualisation pour que nous évitions de voir la main pénétrer dans le volant. Pour cela nous allons utiliser l'idée évoquée déjà précédemment. La plupart des objets peuvent être représentés ou enveloppés dans une sorte de combinaisons d'objets définies mathématiquement par des fonctions implicites, comme par exemple des « superquadriques ». Une représentation implicite consiste à définir une surface ou un volume sous la forme d'une équation de type  $F(x,y,z) \leq k$  avec  $k$  étant une constante,  $F$  la fonction de potentiel. L'idée est de construire en même temps que la scène virtuelle une scène simplifiée permettant de gérer plus efficacement les collisions entre les objets.

$$F(x,y,z) = \left( \left( \frac{x}{a_1} \right)^{\frac{2}{\epsilon_2}} + \left( \frac{y}{a_2} \right)^{\frac{2}{\epsilon_2}} \right)^{\frac{\epsilon_2}{\epsilon_1}} + \left( \frac{z}{a_3} \right)^{\frac{2}{\epsilon_1}}$$

$a_1, a_2, a_3$  : facteur d'échelle  
 $\epsilon_1$  : courbure latitudinale  
 $\epsilon_2$  : courbure longitudinale

Avec l'utilisation des fonctions implicites, nous pouvons calculer plus rapidement les points de collisions. Les contacts et pénétrations entre l'objet virtuel, plutôt les primitives géométriques qui l'englobent (parallélépipède, « superquadrique », etc..) et la main virtuelle sont testées au niveau des senseurs placés à chacune des articulations des doigts et leurs extrémités. On peut aussi augmenter leurs nombres pour améliorer la saisie. Il nous reste maintenant à calculer le point de collision pour chaque senseur. Lors de la manipulation, chaque senseur de la main bouge d'un point P1 vers un point P2 de l'espace. Si P2 intercepte l'objet 3D, cela veut dire qu'une zone de la main est en collision. Il nous faut calculer le point

de collision situé à la surface de l'objet, permettant ainsi de corriger la main par rapport à l'objet, ainsi que les forces de réactions et de répulsion. Tout d'abord nous devons nous concentrer sur le calcul de l'intersection du segment  $\overline{P1P2}$  par rapport à l'objet que nous manipulons. Nous disposons de plusieurs tactiques. Prenons tout d'abord un exemple simple celui de l'intersection d'une droite et d'un ellipsoïde.

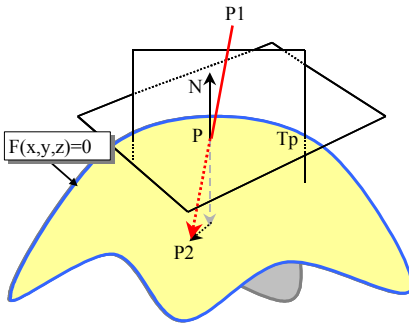


Figure : P1 représente la position d'un des senseurs de la main à l'instant t1 et P2 à t2. Nous devons déterminer P pour pouvoir ajuster correctement le contact de la main sur l'objet. De plus comme le calcul se fait à chaque instant  $\Delta t$  la distance entre P1 et P2 est petite, ce qui fait que la convergence de l'algorithme vers le point de collision est rapide.

Alors, soit  $P(x,y,z)$  le point d'intersection entre la droite  $P1P2$  et la « superquadrique »  $F(x,y,z)$  ;  $\Delta R$  le rayon de la sphère autour du point P ;  $\epsilon=0.00001$  une condition d'arrêt

```

B=Vrai;
Tant que B=Vrai faire
{
    P=(1/2)*(P1+P2)
    v=F(P,ΔR)
    Si (v<0) alors P2=P sinon P1=P
    Si (v≥0) &(v<ε) alors B=Faux
}

```

Remarque : dans le cas où P1 et P2 se trouvent dans la « superquadrique », on calcule l'équation de la droite et on fixe P1 en dehors de la « superquadrique », puis on relance l'algorithme.

Dans notre système, nous disposons désormais d'un moyen pour déterminer le point d'intersection entre la trajectoire décrite par un des senseurs de la main virtuelle et l'objet

manipulé. Il reste à corriger la disposition des doigts sur l'objet à partir de l'information des senseurs et déterminer les forces [Boulic] dans le cas où l'on utiliserait un dispositif hardware à retour d'effort.

L'aspect de la non visualisation des défauts visuels est primordial, certes, mais il n'y a pas que cela. Il faut que l'utilisateur se sente à l'aise dans la manipulation des objets virtuels qui l'entourent. Il ne doit pas se poser des questions sur comment saisir l'objet en question. Le système doit superviser et contrôler ses actions et corriger la disposition des doigts autour de l'objet.

Il faut aussi remarquer qu'il y a une différence non négligeable entre les postures de la main réelle et virtuelle, car les informations provenant par exemple des « *DataGlove* » sont insuffisantes, c'est pourquoi nous extrapolons par des algorithmes et des relations entre les phalanges, les postures pour pouvoir approcher au mieux de la réalité.

De plus, dans le cas où il n'y aurait pas de retour d'effort, nous devons appliquer le principe du « *Ghost Hand* ». L'utilisateur quand il serre le poing et que ses doigts serrent déjà l'objet virtuel, ses doigts ne pénétreront pas dans l'objet.

Maintenant, il nous reste à corriger le positionnement des doigts de la main virtuelle autour de l'objet virtuel. Pour cela, nous disposons de deux approches, l'une de manière indirecte c'est-à-dire par la cinématique inverse et l'autre de manière directe basée sur une recherche dichotomique de la solution combinée avec les informations de l'état de la posture de la main dans le passé et des propriétés angulaires des doigts. Nous expliquerons en détail un peu plus loin.

Connaissant les points de collisions localisées sur la « superquadrique », nous pouvons grâce à la cinématique inverse [Baerlocher 2004] déterminer les angles des doigts et faire les corrections nécessaires. La cinématique inverse qui est une technique pour contraindre et contrôler le mouvement. Elle permet de contrôler les axes de la chaîne en partant de la position que l'on désire atteindre que l'on nomme « *effecteur* ». La posture dépend donc de

l'état courant des angles des articulations de la chaîne. Le problème se met sous forme de système d'équations non linéaire définissant le positionnement de l'effecteur en fonction des angles de la chaîne articulée représentant le modèle cinématique directe. La résolution du système peut conduire à une non solution : (l'objectif à atteindre n'est pas dans la zone de couverture possible, ou à une redondance de solutions possibles). Pour répondre au problème et déterminer la solution acceptable, il suffit de tenir compte des postures effectuées par l'utilisateur juste avant.

L'autre méthode plus directe est basée sur l'idée suivante. Pour chacun des doigts nous connaissons, la position des senseurs, juste avant collision ou pénétration, ainsi que les points de collision sur l'objet. Avec ces informations, nous pouvons appliquer la méthode suivante. On part du point senseur1. Si le senseur1 interpénètre l'objet, on translate la position de la main sur le point de collision. Puis, nous testons après détection de collision bien sûr si le senseur 2 pénètre l'objet ou non si oui on incrémente ou décrémente l'angle par interpolation entre l'instant  $t1$  et  $t2$ , sachant l'information qu'à partir de la position de collision sur l'objet on peut déterminer le coefficient d'interpolation. On applique cette méthode pour les senseurs suivant. Nous pouvons voir les résultats sur la figure ci-dessous.

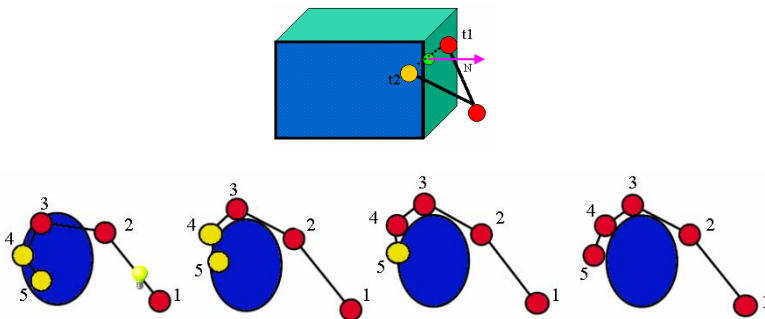


Figure 25 : Repositionnement des senseurs autour de l'objet.

Pour éviter de faire des calculs et des tests inutiles suivant le mouvement, nous utilisons les relations angulaires pour accélérer l'algorithme et les renseignements concernant une collision éventuelle précédente dans le sens croissant ou décroissant des angles. On peut ainsi bloquer la mise à jour de la main virtuelle à tester, ce qui évite des calculs de collisions inutiles. On peut ajouter un second test qui consiste à déterminer si l'objet est saisi ou non, ce



qui permet fixer la posture de la main sur l'objet et faire tourner l'objet autour d'un axe de rotation dépendant de la rotation de la main.

Suite à cette réflexion nous avons donc implémenté cette méthode pour notre simulateur de voiture. Nous avons donc dans le cas d'un volant, décomposé cet objet sous la



Figure 26 : Interaction avec le Volant via un avatar humain

forme d'un tore, de 3 cylindres et d'un ellipsoïde. Ces objets primitifs ont des caractéristiques géométriques dont on connaît leurs équations (équation implicite, équation paramétrique, et...). Notre système de collision utilise ces objets mathématiques et va donc nous permettre de corriger la disposition des doigts autour de l'objet que nous voulons

saisir et manipuler. De plus, la main et l'avant bras de l'avatar humain ont été

traduite géométriquement par utilisation de cylindres à base ellipse. Une décomposition main-bras en 30 et 10 points de contact, situés pour certain au niveau de l'articulation ont été nécessaire.

A chaque instant de la simulation, on projette tous les points « senseurs » de la main sur le ou les objets les plus proches. Ce qui permet de dégager une foule de renseignements (vecteur normal, vecteur pénétration, vecteur force, qu'elle est la partie en collision, et...), de connaître le point de contact, et la distance entre le point « senseur » moyennant une petite correction (car il faut tenir de l'épaisseur du membre). Si une partie de la main est en collision, alors on corrige la disposition de la main par l'étude de son mouvement jusqu'à ce que la distance extérieure soit la plus petite possible.



Figure 27 : Disposition des senseurs de l'avatar humain

Dans le cas où on n'aurait pas de retour d'effort mais uniquement une collision, on corrige la position de la main virtuelle tant que l'on est en collision. Par exemple, si on effectue un mouvement de droite à gauche avec notre bras et qu'au milieu du parcours ce trouve un poteau virtuel, les parties en collision seront corrigées et stoppées. Le bras virtuel au niveau de la vision n'a pas l'autorisation de pénétrer dans le poteau. Et dans le cas où on serait muni d'un exosquelette avec retour d'effort, le mouvement accompli par l'utilisateur serait stoppé.

Concentrons-nous maintenant sur une autre problématique celle de l'action de la main sur un objet. On distingue alors deux types de manipulations avec ou sans contraintes. Si on manipule une balle c'est simple, mais par contre changer la position d'un levier de changement de vitesse ou tourner un volant c'est moins évident, car ces objets sont astreints à

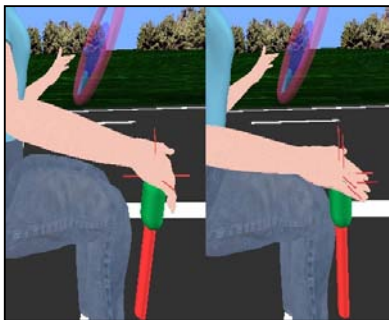


Figure 28 : Manipulation du levier de vitesse par l'avatar humain

se mouvoir dans des positions et directions distinctes. Les étapes sont de déterminer la posture de la main, de déterminer la densité de collision, d'analyser le mouvement que l'on fait, d'effectuer les modifications nécessaires sur le ou les objets, puis de corriger le positionnement de notre membre supérieur en utilisant un calcul d'une posture probable (utilisation de la cinématique inverse pour le positionnement du bras et de l'avant bras, ou recherche de position ciblée c'est-à-dire on donne la position du

poignet, et enfin on effectue la correction). Ce qui a été développé tout au long, ce n'est pas une procédure de « *grasping* » automatique, car on se sert de l'information de la posture de la main réelle jusqu'à ce que l'information de collision l'emporte. C'est-à-dire, lorsque les doigts de la main se referment sur l'objet virtuel, les doigts virtuels sont arrêtés au fur et à mesure qu'il détecte une collision. On peut avoir le poing fermé, tandis que la main virtuelle saisissant l'objet est semi-ouvert.



Figure 29 : Calcul des collisions entre l’avatar piloté par l’utilisateur et les objets primitifs (tore, cylindre, ellipsoïde) qui composent et englobent le volant, et le levier de changement de vitesse.

Finalement, nous pouvons dire que manipuler des objets ayant une structure complexe (comportant beaucoup de meshes, comportant des formes compliquées) en temps réel avec une qualité proche du réel est loin d’être évident. Il faut trouver des méthodes de détection de collisions rapides et qui restent efficaces quand le nombre ou la complexité des objets simulés croît. Notons qu’elle peut dans certain cas accaparer plus de 90% du temps de calcul. Des méthodes d’accélération pour détecter des zones de collisions probables sont alors nécessaires. Les approches sont très diverses et nécessitent souvent d’être combinées pour un maximum d’efficacité. La simulation physique reste coûteuse et accapare beaucoup de ressources. En conséquence, nous pensons que pour interagir en temps réel, il faut utiliser des objets simples (sphère, cylindre, tore,...), non compliqués, ce qui permet ainsi de réduire les coûts de calculs de collisions.

### 3.4. DISCUSSION

L’analyse des données récoltées dans nos différents tests d’évaluation sur les différentes interfaces, nous a permis d’observer les résultats suivants.

Nous avons pu observer, d’une part que lors de nos tests de conduite certaines personnes étaient victimes de maux ou de nausées : le « *cybersickness* » (terme proposé par McCauley et Sharkey), phénomène tout à fait normal observé dans de nombreux simulateurs car c’est une réponse renforcée par le degré de sensation d’immersion dans le monde virtuel à un ou des stimuli inhabituels. Nous pouvons diminuer ces effets par un jeu de subtil de combinaisons diminuant ainsi les contradictions des informations arrivant à notre cerveau.

Une bonne étude a été faite sur la « *cybersickness* » par le chercheur J. LaViola qui a fait le point sur les éléments morphologiques qui entrent en jeu, les conséquences psychologiques du conflit sensoriel, ainsi que les facteurs aggravants qui l'engendrent. [LaViola2000] Les problèmes liés peuvent provenir de trois raisons [Chevennement1998] le profil de l'utilisateur, les caractéristiques du simulateur (dynamique, statique, temps de latence éventuels, retours kinesthésiques) et du protocole d'essai (charge mentale). Par exemple, les personnes ayant une faible expérience de la tâche réelle seront moins malades que celles qui connaissent bien la tâche. En effet, leur référentiel est moins important. En fonction de ces caractéristiques, il peut y avoir une contradiction auriculo-vestibulaire/vision et des problèmes de convergence, mais ceci est vraiment dû aux caractéristiques des appareils de la réalité virtuelle. Ces problèmes sont dus en grande partie au temps de latence dans l'affichage. Ils sont surtout perçus lors d'une manipulation précise ou lors de l'utilisation d'un visiocasques. Ce temps de latence dans l'application introduit un délai entre une action dans le monde réel et son équivalent dans le monde virtuel. Ce délai lié à l'acquisition des données venant des capteurs peut augmenter si l'on doit passer un filtre pour gommer le bruit ou si la scène à afficher est complexe.

Nous pouvons dire qu'il existe un seuil pour lequel le phénomène de latence devient perceptible pour certaines personnes et contribue aux symptômes diagnostiqués dans la plupart des simulateurs. C'est pourquoi nous nous sommes efforcés de réduire au maximum ces temps de latence. Un travail d'optimisation des différents algorithmes a donc dû être fait.



Figure 30 : Conduite de la voiture sous HMD

Autre fait majeur constaté c'est que dans le système CAVE en mode stéréovision, les candidats avaient du mal à estimer les distances entre leurs mains réelles et les différents outils de conduite virtuels du véhicule. La saisie d'un objet virtuel avec une main réelle sous contrainte d'un exosquelette avec retour d'effort était difficile pour certaine personne par contre les experts trouvaient facilement l'emplacement du volant et le levier de changement de vitesse virtuel. Ceci étant peut être du à une mauvaise interprétation, et à une mauvaise représentation mentale des choses. Il faut garder en tête que lorsqu'on conduit une vraie voiture, on ne passe pas les vitesses en regardant le dispositif. Donc, sentir le levier de changement de vitesse cela aide. L'expérience nous a montré que l'utilisation de ce système nécessite un certain temps d'adaptation qui peut être plus ou moins long suivant la capacité des individus. Il existe donc une subtile combinaison entre le visuel et le tactile.

Concernant le système HMD, l'utilisateur est complètement immergé dans son environnement virtuel et se voit représenté par un avatar, un humain virtuel esclave plus ou moins des mouvements de l'utilisateur. La représentation de l'avatar fournit donc les réactions visuelles nécessaires pour s'identifier à lui. L'utilisateur conduit par l'intermédiaire de l'avatar la voiture. Nous utilisons notre système de détection pour agir sur les objets.

Les expériences réalisées nous ont indiqué qu'il n'était pas possible de conduire sans dispositif à retour d'effort car cela demandait trop d'attention au conducteur. C'est trop difficile, voire impossible d'interagir et de conduire la voiture en faisant attention aux autres voitures. Certains conducteurs vont vite dans le décor. Un dispositif à retour d'effort est donc indispensable.

Autre fait marquant observé, c'est que le champ de vision limité dans le HMD aggrave la situation de conduite. L'information visuelle est restreinte. Le conducteur ne peut pas anticiper le déplacement des autres voitures. Pour faciliter la conduite, nous avons alors introduit un concept de réaction passive [Kallman2003], c'est-à-dire le principe de la conduite automatique par des gestes simples. Changement automatique des vitesses de conduite lorsque l'utilisateur appuie sur la pédale des gaz. Les mains virtuelles sont remplacées par des sphères comme indicateur de position. L'utilisateur interagi avec les périphériques de contrôle en faisant coïncider les sphères dans les zones d'activation des dits périphériques. Nous observons que les résultats au niveau de la conduite sont meilleurs, mais il y a tout de même des difficultés à faire coïncider l'espace virtuel et l'espace réel. La précision des traqueurs en

est pour quelque chose. L'utilisateur a quelque fois du mal à trouver les zones d'activations, à cause du manque de précisions.

Finalement de multiples raisons nous ont conduits à l'utilisation d'un exosquelette avec retour d'efforts. Beaucoup d'applications exigent que le système soit capable de simuler en temps réel des sensations tactiles pour atteindre le réalisme souhaité (ressentir et percevoir les objets rigides et déformables). De plus d'après les travaux de Hannafort and Wood (1989) dans le contexte de la « téléopération », ont montré que l'utilisation d'un périphérique à retour d'effort réduit le temps de l'accomplissement d'une tâche de 30%, réduit de 60% les erreurs de manipulations.

Nous pouvons aussi nous focaliser sur les points suivants : Les utilisateurs ne doivent pas être encombrés par l'équipement hardware. Si l'utilisateur interagit dans un environnement virtuel sans contrainte, aucune force perceptible ne doit lui être renvoyée lorsqu'il y a interaction ou mouvement. Les objets virtuels définis comme rigides doivent être perçus comme rigides. Donc, l'un des paramètres d'un système à retour d'effort est la dureté maximale de la surface des objets virtuels qu'il est capable de représenter. Si l'utilisateur peut ressentir qu'une modélisation d'un mur virtuel est un mur, le système est susceptible d'être efficace, encore faut-il qu'il le soit lors d'un déplacement d'une balle par exemple. Une certaine cohérence entre ce que l'on voit et ce que l'on ressent est nécessaire. Il n'y a rien de plus perturbant que de traverser un objet rigide virtuel. Nous devons contraindre et bloquer les mouvements de l'utilisateur. Des forces doivent lui être appliquées, encore faut-il déterminer les différentes valeurs. La force maximale exercée par les doigts d'une main humaine est de l'ordre de 40 Newton, mais durant une opération précise, nous pouvons nous contenter de 10 Newton d'après les mesures observées. Concernant le blocage des membres supérieurs, les bras, 100 Newton sont nécessaires si on veut par exemple interagir avec le volant virtuel et le levier de changement de vitesse. L'utilisateur doit pouvoir s'appuyer et ressentir ces périphériques de contrôle virtuel. C'est vraiment un paramètre important pour augmenter le facteur présence dans notre environnement virtuel [Hoffman1996]. On peut aussi augmenter l'immersion avec d'autres moyens pour isoler la personne de la réalité [Scuemie1999].

Par conséquent, beaucoup de compagnies ou d'universités comme nous l'avons vu dans le chapitre précédent développent de nouveaux design et interface « *haptique* » pour la

main, pour le bras ou les deux par exemple [Frisoli2002], [Gosselin2005] et [Immersion2003].



Figure 31 : Exosquelette de PERCRO ; Le WHIPFI prototype du CEA ; La « *station haptique* » d'IMMERSION™

Jusqu'à maintenant la technologie de l'« *haptique* » se focalisait toujours sur les mains plus que d'autre partie du corps. Beaucoup de chercheurs cherchent à trouver la meilleure façon de réaliser une interface qui soit la plus naturelle possible, mais pour l'instant il existe une barrière peut-être infranchissable. C'est une barrière physique entre les capacités sensorielles humaines et la technologie hardware des systèmes d'interactions. C'est pourquoi la recherche devrait se focaliser dans la direction de système intuitif. [Stone1993]. Si une interface de « *haptique* » n'est pas utilisée de manière intuitive, les notions de présence et de croyance dans l'action s'amenuisent car l'opérateur peut avoir des difficultés, sources de stress et de fatigue.

Les technologies basées sur l'« *haptique* » souffrent de plusieurs limitations qui vont au-delà du prix des interfaces « *haptiques* » développées. Elles sont loin d'être concluantes. Ces inconvénients incluent la limitation de l'espace de travail, le poids des appareils, le manque de forces de réaction sur les membres de l'utilisateur, et les problèmes de calcul de collision en temps réel permettant une bonne restitution des forces. Comme par exemple, dans le domaine de la télémanipulation : la « nanomanipulation » [Finch1995], le problème est de simuler correctement la surface et les forces suivant un rapport d'échelle entre les deux univers. Dans le domaine des applications chirurgicales (modélisation des tissus) [O'Toole1997], le problème est de simuler les forces qui seraient senties réellement par le chirurgien lors d'une opération. Comme autres inconvénients, nous pouvons évoquer ceux liés à l'ergonomie des appareils causant dans la plupart des cas gênes et fatigues de l'utilisateur.

Une miniaturisation est donc exigée. Il faut aussi prendre en considération les particularités du problème par rapport à l'application spécifique.

Par exemple dans le cas de la « *station haptique* », cet outil présente des limitations, comme la direction des forces exercées sur les doigts de la main qui s'effectuent uniquement dans la même direction, le poids des parties métalliques qui rendent inconfortable les mouvements, les câbles courts qui relient le gant à son utilisateur, et la nécessité d'utiliser cet appareil avec un « *DataGlove* » pour être efficace. En dépit de tout cela, cet appareil est très puissant et générique pour les manipulations manuelles.

### 3.5. CONCLUSION

Ce chapitre nous a permis donné un bref aperçu par l'intermédiaire d'une application concrète, explorant plusieurs champs de domaines, des contraintes et des difficultés rencontrées pour développer un simulateur de voiture. Les résultats obtenus par les tests sur des candidats nous ont fournis de multiples informations qui nous permettrons de corriger et d'améliorer le système. Cette étude nous a permis d'étudier le comportement interactif des utilisateurs par rapport aux différentes interfaces. Tout au long du développement de l'application nous avons fait des choix conditionnés par des contraintes terribles : le calcul en temps réel et le réalisme.

Nous avons été alors confrontés à des problèmes relatifs au temps, c'est à dire d'une part les temps de latence : délais entre les différentes boucles (moteur de collision, moteur de calcul de force, communication entre les machines) et d'autre part la synchronisation des tâches réparties sur les différentes machines (Exemple le système CAVE).

Nous avons aussi rencontré des problèmes de calibrations entre le monde réel et virtuel. Dans le cas de notre simulateur de conduite (système CAVE), nous devons ajuster de manière précise les mouvements de notre exosquelette par rapport à la voiture virtuelle. L'exosquelette joue en effet deux rôles : il contraint les mouvements de l'utilisateur et contrôle les mouvements de l'avatar humain même s'il n'est pas toujours affiché. L'utilisateur se glisse dans la peau de l'avatar humain tel un gant pour pouvoir conduire la voiture. Mais,



étant donné que les morphologies des personnes ne sont pas les mêmes et que l'exosquelette déplaçable vient se mettre autour du membre supérieur de la personne, nous avons dû mettre des capteurs de positionnement sur l'exosquelette. Au niveau des « *DataGloves* », il fallait que le positionnement et les mouvements des différentes parties de la main virtuelle coïncident parfaitement avec les mouvements de la main du conducteur. Un certain réalisme gestuel des doigts entre la main réelle et la main virtuelle était nécessaire car une mauvaise calibration nous conduisait naturellement à des difficultés d'interaction avec les éléments internes de la voiture virtuelle. La calibration est une étape qu'il ne faut en aucun cas négliger.

Un autre problème constaté, est celui de l'estimation des distances. Pour résoudre le, il a fallu essayer de calibrer la stéréovision pour l'adapter à la personne (*vous estimez cet objet à quel distance ? et on adapte en fonction de la réponse donnée*) et d'améliorer le réalisme de l'image en travaillant sur les ombres. L'information « ombre » est un bon indicateur involontaire car nous nous servons naturellement de cette information sans être conscient.

Et enfin des problèmes de définition du système interactif avec l'utilisateur ; de calcul de la restitution sensorielle du système simulé ; de réalisme du comportement sensoriel ; de réalisme du comportement autonome des éléments composants une ville.

Voilà, donc les problèmes auxquels nous avons été confrontés et qui nous ont permis d'approfondir notre réflexion et de finaliser cette idée conceptuelle d'avoir la possibilité de changer très rapidement l'ergonomie de la voiture avec tout le confort « *haptique* » souhaitable pour augmenter la qualité de l'interaction. sont des concepts importants qui réduisent les coûts de fabrication de prototypage du véhicule. L'utilisation d'un système « *haptique* » avec retour d'effort comme par exemple une « *station haptique* » d'IMMERSION™ ou l'exosquelette de PERCRO combiné à un dispositif d'immersion totale permet de nous faire l'économie de développer de multiple plate-forme de simulateur spécifique à un prototype de véhicule donné. L'avantage d'un tel système est que tout est paramétrable et adaptatif aux désirs de l'utilisateur. Encore faut-il disposer d'un bon système software et hardware qui s'approche très bien de la réalité, ce qui n'est pas encore le cas aujourd'hui, mais grâce au progrès nous tendons rapidement vers cette réalité. Les progrès technologiques, miniaturisation, confort, qualité tant au niveau hardware que visuel, ainsi que la compréhension des mécanismes faisant naître en chacun de nous les nausées dans une simulation, vont contribuer à diminuer les symptômes perçus par les utilisateurs.

Au jour d'aujourd'hui, nous devons prendre en compte toutes les remarques observées jusqu'à maintenant, car faire travailler une personne en immersion totale est loin d'être une tâche aussi facile. La recherche dans le domaine de l'« *haptique* » est loin d'être terminée, il reste encore beaucoup de choses à faire pour que les utilisateurs interagissant dans un environnement virtuel croient à ce qu'ils voient à ce qu'ils ressentent tout comme dans le réel. Une interface « *haptique* » efficace exige la restitution fidèle des phénomènes « *haptiques* », ce qui requiert un support matériel et une approche de conceptions spécifiques. Nous croyons que les interfaces « *haptiques* » offriront de meilleurs moyens pour interagir naturellement avec le monde 3D de manière intuitive. Mais le développement de ces technologies, ainsi que des méthodes appropriées, sont de vrais challenges dans la recherche en réalité virtuelle.

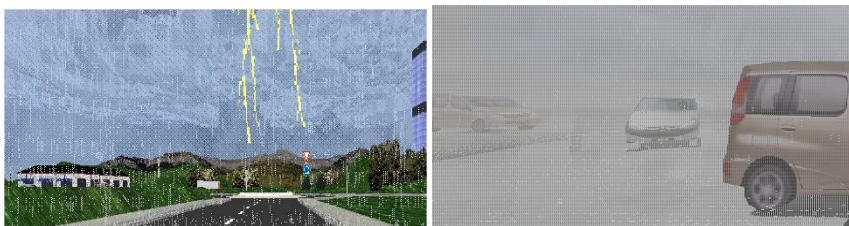


Figure 32 : Vues de VIRTUAL.

Dans VIRTUAL, il est possible de changer les conditions météorologiques afin de perturber un peu plus le conducteur.



Figure 33 : Illustration de piétons marchant dans une ville

Pour tout complément d'informations concernant le projet VIRTUAL s'adresser au  
Centro Ricerche FIAT, Orbassano/Turin, Strada Torino, 50, I-10043 Orbassano (TO), Italy  
Email : Cecilia RUSPA [cecilia.ruspa@crf.it](mailto:cecilia.ruspa@crf.it)

## CHAPITRE 3 : LE CONCEPT DU MEDIATEUR

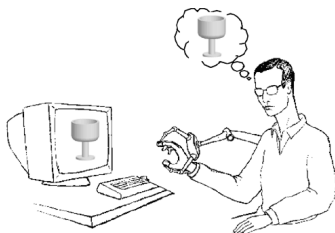
L'objectif de ce chapitre est de présenter les grandes lignes du concept du médiateur qui pourrait s'expliquer de la manière suivante : Un utilisateur plongé en immersion totale dans un univers nommé monde médiateur, va pouvoir agir à distance, par l'intermédiaire d'un périphérique « *haptique* », sur un autre monde virtuel ou réel nommé monde contrôlé.

Tout au long de chapitre, nous allons décrire ce qui nous a conduit à la définition de ce concept et nous allons mettre les problèmes à plat afin de proposer une interface idéale permettant d'interagir de manière efficace avec des entités dites complexes en définissant aussi clairement l'environnement de travail, les ressources, les besoins.

Enfin après réflexion, nous ferons des propositions sur une structure d'interaction visuelle entre l'utilisateur, en immersion totale ou « semi-immersion », et son environnement.

### 4.1. LE CONCEPT

Nous avons vu jusqu'à maintenant que la manipulation d'objets complexes demeure compliquée. Certaines manipulations d'entités (objets virtuels, robots,...) requièrent donc l'utilisation d'intermédiaires simples et adéquats, dénommés médiateurs, simplifiant ainsi l'interaction entre l'homme et l'entité.



Avant toute chose, on peut se poser les questions de l'opportunité d'utiliser un intermédiaire et des risques d'altération des relations qui s'établissent entre l'individu et son entité qu'il désire manipuler.

Comme on le sait, lorsque nous manipulons des objets dans le monde réel, les relations visuelles et tactiles nous permettent de construire des représentations mentales.

Ces multiples informations établissent un lien entre l'objet et l'être humain. La qualité de la transcription (transposition) de l'objet dans le cerveau conditionne l'analyse des possibilités d'actions et les décisions de choix. Que pouvons-nous faire avec cet objet ? Que va-t-on faire ? Comment allons-nous procéder ?

Si l'on désire faire des modifications sur un objet mais que l'on ne dispose pas les moyens nécessaires pour modifier son état ou son statut, nous devons utiliser un outil. Cet outil est-il un intermédiaire ou un moyen pour l'utilisateur d'accomplir sa tâche et d'atteindre son objectif ?

On peut prendre l'idée d'un sculpteur qui utilise des burins comme outils. Des relations s'établissent entre lui et son bloc de pierre. L'envie de création a suscité en lui des idées et des objectifs intermédiaires de réalisation, de façonnage et de modelage. En contact avec la matière, les relations entre l'homme et sa sculpture sont directes.

Si nous prenons maintenant le cas, d'une personne qui veut déplacer un objet très lourd. Du fait de son poids, elle va devoir utiliser un intermédiaire pour pouvoir le déplacer. Les relations entre la personne et l'objet à déplacer ne sont plus les mêmes, car elle va devoir agir indirectement et à distance pour l'action.

Autre exemple, celui du musicien. Le musicien ne peut en aucun cas produire par lui-même le son qu'il désire jouer. L'instrument de musique est un médiateur entre lui et le son qu'il désire produire. Dans ce cas également, il y a une certaine distance relationnelle avec son entité.

Dans un autre domaine, celui de l'informatique et de la « téléopération », nous avons vu que la manipulation directe d'objets complexes est un peu trop compliquée car nous ne disposons pas encore des techniques nécessaires pour une parfaite transcription du monde réel dans le monde virtuel (problèmes au niveau « *haptique* » et au niveau des interactions directes). L'utilisation d'outils médiateurs, plus simples à utiliser serait peut-être judicieuse pour contourner le problème en permettant une interaction indirecte.

L'outil médiateur est un moyen de communication simple entre l'utilisateur et l'action qu'il doit réaliser sur l'objet. Nous pouvons dire qu'il y a une répartition des mécanismes de réalisation de la tâche d'une part entre l'humain et son médiateur et d'autre part entre le médiateur et son entité contrôlée.

Mais il ne faudrait surtout pas que lors de la réalisation d'une action, l'attention de la personne se porte plus sur l'entité médiatrice que sur l'entité contrôlée. L'utilisateur doit faire abstraction de l'outil utilisé ou faire corps avec lui. Il doit focaliser sa pensée uniquement sur l'entité manipulée comme nous le faisons avec la souris de notre PC.

Ainsi des relations proches mais non directes sont établies entre l'opérateur et son entité contrôlée. C'est donc cette idée qu'il va falloir mettre en pratique. Le médiateur est un dispositif simplifiant les opérations compliquées à réaliser.

Dans notre concept, nous distinguons alors deux mondes. Le monde médiateur contenant les entités médiatrices et le monde contrôlé contenant les entités que l'utilisateur désire manipuler. Ces deux mondes peuvent coexister ou être distincts suivant l'application. L'idée que nous proposons ici est qu'un utilisateur en milieu plus ou moins immersif, utilisant par exemple un périphérique à retour d'effort, interagit de manière indirecte sur un ou des entités du monde contrôlé via une interface médiatrice permettant de manipuler l'entité de manière simple à l'aide de périphériques virtuels nommés entités médiatrices.

Notre choix c'est finalement porté sur un milieu immersif car ce dispositif permet à l'homme de se plonger dans un monde sans relation avec le monde réel. De plus, le degré d'immersion dépend plus ou moins de la quantité et de la qualité des informations transmises à l'utilisateur. L'utilisation d'un environnement virtuel nous permet de construire plus rapidement des entités médiatrices adaptables suivant le profil de la personne, ce qui n'est pas le cas dans le monde de la télémanipulation.

Suite à toutes ces remarques, nous pouvons schématiser notre concept du médiateur par la représentation suivante :

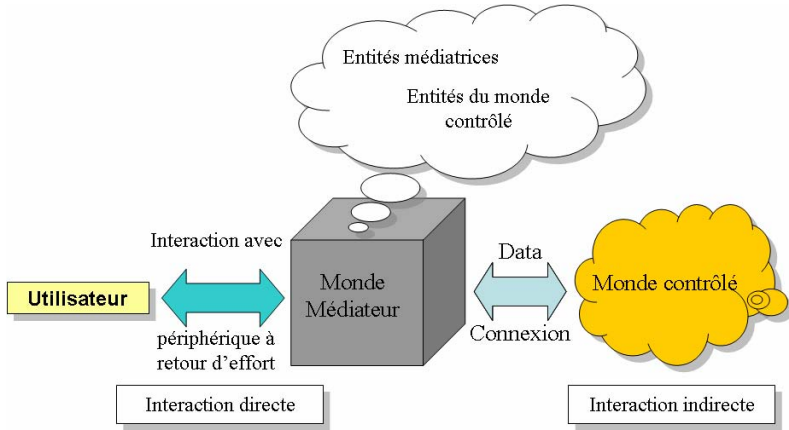
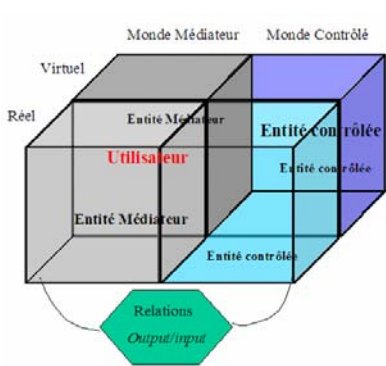


Figure 34 : Représentation schématique du concept.

Grâce au monde médiateur, l'utilisateur interagit de manière indirecte sur l'entité du monde contrôlé.



Cette idée de médiateur offre quatre possibilités d'interaction. Un monde médiateur virtuel peut interagir avec monde contrôlé virtuel comme c'est le cas de l'application des menus 3D ou celle permettant de contrôler un robot par l'intermédiaire d'un tapis roulant (*voir chapitre suivant*). Le monde contrôlé est réel et contient le robot. L'utilisateur en « semi-immersion » dirige le robot en utilisant une interface réelle. Ce système laisse l'opérateur marcher derrière

un robot virtuel projeté sur un écran représentant le robot réel. C'est une conduite à la troisième personne ; concept déjà utilisé dans plusieurs jeux vidéo. L'interface médiatrice est donc dans notre cas : un monde « semi-immersif » contenant une représentation de notre robot, de l'image Webcam du robot mobile réel et d'un tapis roulant comme interface entre l'utilisateur et son application. Et puis marcher derrière un robot en utilisant le principe du caddie, c'est un moyen plus naturel que d'utiliser un « joystick ».

Il semble intéressant d'utiliser ce concept dans le cadre de la « téléopération » en fournissant à un utilisateur une interface qui lui permette de contrôler, de gérer, de changer les propriétés, de manipuler efficacement des entités d'un monde virtuel ou réel via un monde médiateur purement virtuel.

L'utilisateur interagit indirectement sur l'objectif désigné au moyen d'outils virtuels qu'il utilise directement, comme on le fait en « téléopération ». Le choix de l'interaction est tout indiqué car on sait que les interfaces physiques (réelles) développées pour la télémanipulation sont cher et difficiles à reconfigurer.

Reste encore à injecter dans cette interface des sensations « tactilo-kinesthésiques ». Les technologies « *haptiques* » à retour force sont là pour résoudre ce problème. On peut aussi fournir à l'utilisateur une structure lui permettant de construire des interfaces virtuelles adaptatives pouvant être reconfigurées librement afin de contrôler un large panel d'applications.

Une notion de flexibilité est donc à introduire en ce qui concerne l'ajustement de l'interface de commande qui peut-être soit automatique (modélisée par l'utilisateur ou interfaçage intelligente) ou soit manuelle (modifiable sur intervention de l'utilisateur au niveau alias et fichier de profil). Telle est notre idée.

Maintenant nous allons voir ce qui nous a conduit à cette conceptualisation, puis nous le développerons le concept d'interface médiatrice plus en détail.

## 4.2. JUSTIFICATION

Aujourd'hui, la capacité de toucher et de manipuler des objets virtuels en utilisant des modèles naturels de mouvement de la main est une exigence fondamentale pour le développement futur de nouvelles technologies de l'environnement virtuel.

Notre premier objectif était de développer des outils permettant d'interagir en temps réel avec des objets, à l'aide de périphériques hardware (« *gloves* », exosquelettes avec ou sans retour de forces), tout en maintenant une cohérence la plus réaliste possible. L'utilisateur devait avoir la sensation lors de la manipulation de certains objets (sous « *Head-Mounted-Display* ») que sa main virtuelle était sa vraie main.

Compte tenu de tous les problèmes observés et évoqués dans les chapitres précédents, tant au niveau hardware, qu'au niveau calcul de collisions, nous ne pouvions pas continuer sur cette voie de recherche et nous nous sommes dirigés vers l'utilisation de méthodes de manipulations indirectes peut-être plus efficaces.

Il fallait donc introduire dans notre monde le concept de manipulation indirecte pour être capable d'agir sur les objets plus confortablement. Notre première idée a été d'avoir recours au concept des menus 3D ou « widget 3D » puisque chaque item d'un menu réalise une tâche bien définie.

Mais il faut savoir comment les utiliser, comment les disposer dans la scène virtuelle sans altérer l'interaction, comment désigner efficacement ce que l'on désire sélectionner et comment interagir efficacement de manière indirecte sur des objets 3D en milieu immersif ou « semi-immersif ». Les réponses à ces questions permettront de proposer des outils qui serviront pour notre interface graphique.

Les expériences nous ont montré que les menus 3D de style classique ne sont pas adaptés pour un environnement 3D. Nous devons donc créer une structure d'un autre type, une sorte de télécommande conviviale facilitant au maximum l'interaction entre l'homme et son interface. Mais revenons auparavant sur les problèmes posés par la sélection.



Toutes manipulations nécessitent de désigner les entités. Un bon système de pointage est donc requis. Le pointage d'un élément est une tâche relativement facile, mais qui demande quelque fois de la concentration dans le cas où nous devons sélectionner des objets lointains ou petits. La désignation de l'objet se fait dans le majeur parti par l'index de la main.

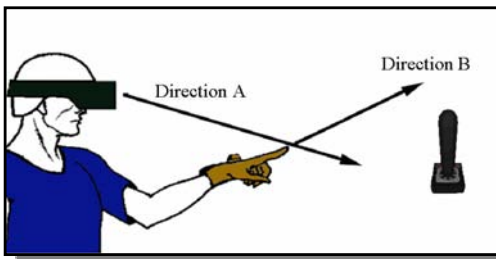


Figure 35 : La désignation d'un objet.

Plusieurs méthodes sont possibles :

Une droite (Direction B) passe symboliquement par le poignet et l'index et on calcule si la droite passe par l'objet désigné ou non. Nous pouvons calculer la direction moyenne entre la droite passant par la tête de l'utilisateur et l'extrémité du doigt (Direction A) et la droite

Direction B. Pour aider à la sélection nous pouvons ajouter un segment de droite au bout de l'index de la main virtuelle de l'utilisateur et mettre des alertes visuelles. On peut utiliser aussi des techniques comme par exemple les « *Skitters and Jacks* » qui sont des extensions du modèle du curseur en croix dans un espace 3D.

Pour améliorer la méthode de sélection, nous pouvons utiliser l'« *haptique* » dans un mode passif ou actif.

**Haptique passif**: Bowman [Bowman1998] utilise une technique basée sur une interaction passive avec les menus 3D, c'est-à-dire d'utiliser comme métaphore d'interaction un stylo et une tablette comme moyen d'aide pour une sélection avec retour de force, combiné à une représentation virtuelle de ces objets dans l'environnement virtuel de l'utilisateur. Les résultats obtenus dans ses expériences ont montré l'efficacité d'utiliser les menus 2D dans un environnement virtuel.

**Haptique actif**: Dans le cas où l'on utiliserait une « *station haptique* » avec retour d'effort, un système d'aide à la sélection basé sur des systèmes de contraintes de forces peut être mis en place. Pour cela il faut changer la physique de l'environnement dans lequel l'utilisateur est plongé ce qui implique la suppression des actions de pesanteur. Un système de

zéro gravité doit être intégré afin de permettre (voir article sur comment compenser) [Ott2005] de diriger la main virtuelle et donc la main de l'utilisateur vers l'objet attractif.

L'environnement est régi par la mécanique de Kepler. L'utilisateur sentira les effets quand il bougera sa main vers la direction de l'objet et dans le cas des menus, il aura la sensation d'être en présence d'un système cranté.

Il faut remarquer que nous utilisons une interaction indirecte pour manipuler plus simplement des entités complexes via des menus 3D, des « *widget 3D* ». Ces entités simples d'utilisation sont des outils médiateurs entre l'utilisateur et ce qu'il désire manipuler. Ce qui rejoint tout à fait l'idée formulée au début de cet écrit qui n'est autre que celle d'une interface médiatrice.

Autre remarque, puisque nous pouvons contrôler par ces moyens des entités complexes, alors il serait intéressant de pouvoir dans le cadre de la « téléopération » de contrôler un robot constituant, selon l'idée du concept du médiateur, une entité complexe dans un monde réel.

En effet, en « téléopération », l'interface de commande est spécifique aux simulateurs donc peu adaptable et peu configurable. Contrôler un robot via une interface 2D laisse à désirer parce que les informations à intégrer par l'utilisateur sont trop compliquées. Le facteur implication n'y est pas et on note une certaine distance entre l'opérateur et la tâche à effectuer.

Grâce à la réalité virtuelle, développée dans le projet **EU VIRTUAL**, nous pouvons transférer le poste de contrôle du monde réel au monde virtuel. Les instruments de commandes sont alors virtuels, ce qui nécessite l'utilisation d'un appareillage à retour d'effort. L'idée est de permettre par la suite à l'utilisateur de pouvoir construire sa propre interface.

Ces remarques ainsi formulées, vont nous conduire à vous proposer une nouvelle structure de notre concept de médiateur.

### 4.3. REFLEXIONS SUR LE CONCEPT

Le médiateur est un concept permettant à un utilisateur d'interagir soit à partir d'un environnement virtuel avec plus ou moins de degré d'immersion sur un ou des entités du monde contrôler. Des relations doivent donc être créées entre les entités médiatrices du monde médiateur et les entités contrôlées du monde contrôlé. Reste à définir comment et par quel moyen ?

Lors de la construction d'une interface, nous devons prendre en considération la notion d' « affordance ». Ce n'est pas suffisant d'avoir à sa disposition une entité médiatrice ; Il faut savoir comment l'utiliser et ne pas se demander ce que l'on pourrait en faire. L' « affordance » reflète les relations possibles entre les acteurs et les objets c'est-à-dire les propriétés du monde. Ce sont des processus définis au sein de l'interface qui naturellement poussent les personnes à anticiper de la façon la plus naturelle possible, les différentes opérations nécessaires pour réaliser les tâches.

Produire de l' « affordance » dans l'interaction est un véritable challenge. Ce qui est simple pour le créateur qui sait évidemment comment il fonctionne ne l'est peut-être pas pour l'utilisateur. L'expérimentation [Geoffrey1997] montre que la compréhension d'une architecture 3D et de son fonctionnement se fait par son utilisation et de manière intuitive. L'utilisateur doit pouvoir avoir la possibilité de la manipuler. C'est évident, mais dans certains cas le côté intuitif n'y est pas, ce qui peut perturber l'utilisateur. Un apprentissage est alors nécessaire.

Prenons l'exemple suivant de l'utilisateur voit un bouton sur un appareil électrique. Il sait immédiatement qu'il doit le pousser ou qu'il doit le tourner pour le faire fonctionner. Ce sont donc des éléments schématiques particuliers qui sont vraisemblablement déjà modélisés dans notre représentation mentale. Il faut noter que la plupart des utilisateurs comprennent l' « affordance » de « widgets » généraux tels que boutons, « scrollbars », etc., Cependant, dès qu'ils sont confrontés à un nouveau moyen interactif, des problèmes se posent. Il faut donc en tenir compte dans l'interface médiatrice car un utilisateur doit toujours savoir quoi faire. Afin que l'interaction soit la plus naturelle possible, il peut être nécessaire de créer un espace isomorphe à l'espace réel.

Cette interface médiatrice devrait être dotée d'outils reproduisant analogiquement toutes les opérations mentales nécessaires à la compréhension humaine. Cela soulève de nombreuses questions : Comment construire ces mondes, Comment permettre l'appréhension de ces mondes aux apprenants ? Comment juger de l'efficacité des outils médiateurs disponibles au sein de ce monde ? Faut-il que l'outil médiateur soit adapté au profil de l'utilisateur ? Faut-il que l'utilisateur soit passif ou actif ? Lors de la mise en relation de deux entités issues du monde médiateur et du monde contrôlé, le système doit-il proposer ou non des solutions ?

Prenons le cas des menus, en fonction du contexte, l'un des items peut venir à nous nous aider à faire le bon choix et pour faciliter sa sélection. Dans le cas d'une sélection directe, l'utilisateur devra donc faire l'effort de concentration pour aller chercher l'information avec tous les problèmes que cela suppose (sélection difficile du à la mauvaise estimation des distances si des métaphores visuelles, sonores ne sont pas utilisés)

Si l'utilisateur a la possibilité de construire sa propre interface, nous devons alors définir correctement les outils nécessaires. Quels types d'outils ? Comment doit-on manager l'interface construite par l'utilisateur ? Le système doit-il proposer divers types de configurations modifiables par l'utilisateur ? L'interface créée par l'utilisateur sera-t-elle efficace ?

Certaines tâches requièrent de se concentrer sur un point particulier, tandis que d'autres nécessitent une vue globale du milieu. En général, l'attention est partagée temporellement entre ces deux cas. L'idéal serait de voir tout ce qui se passe à tout moment. Il y a cependant une limite à la quantité d'information qu'il est possible de traiter et d'afficher que ce soit en milieu non immersif ou immersif. Nous pouvons néanmoins sélectionner une partie de ces informations en s'adaptant à des spécificités et aux limites de la perception humaine.

En particulier, il existe une zone centrale dans le champ visuel où l'image est très précise, alors que la vision périphérique, manquant de netteté, réagit surtout aux mouvements et aux changements. Et c'est pourquoi, nous devons focaliser les principales actions de l'utilisateur dans son centre d'intérêt et gérer d'une manière optimale l'information nécessaire pour l'accomplissement de sa tâche. Mais alors :

*Comment visualiser et répartir l'information visuelle dans notre monde médiateur ?*

*Comment présenter l'information visuelle à l'utilisateur ?*

Nous allons raisonner dans le cas d'une interface médiatrice complexe qui ne se résume pas à un manche à balais virtuel contrôlant un robot réel. Dans notre monde médiateur, nous pouvons contrôler de multiples entités du monde contrôlé et visualiser beaucoup de paramètres. Il est nécessaire de bien présenter les informations afin que le monde médiateur ne devienne pas chaotique et non gérable.

Aujourd'hui, la visualisation de l'information est un champ de recherche en pleine expansion. C'est une présentation visuelle d'informations abstraite dans une structure facilitant leur assimilation rapide et leur compréhension que nous appelons paysages informationnels. (Exemple : un indicateur de positionnement d'une entité contrôlée : un robot sur une carte, ou un indicateur de l'état des ressources en énergie d'un robot).

L'utilisateur peut être immergé dans une multitude de données avec la possibilité d'interagir avec elle, en utilisant bien sûr des outils médiateur. Le but recherché dans l'interaction est l'exploration de données, l'analyse de données, et la présentation des données [Sahling2002]. Ces données peuvent être très complexes, contenir un grand nombre d'éléments qui peuvent être structurés hiérarchiquement, linéairement ou même manquer de structure.

Le but ultime de toute visualisation de l'information est de communiquer des propriétés de l'information à un être humain. Par conséquent, c'est important de considérer l'aspect cognitif. Comment est-ce que le « design » attire l'attention sur certains éléments des données visualisées ? L'utilisateur a besoin souvent de pouvoir concentrer son attention sur certains éléments de l'information visualisée. L'attention humaine est sensible aux différences d'apparence visuelle, telles que les couleurs, les formes, les textures, les ombres projetées, les dimensions. Le mouvement est particulièrement utile pour attirer l'attention de l'utilisateur à la dynamique de la visualisation. L'emplacement d'un objet est également déterminant pour

l'attention. Par contre, les interfaces textuelles exigent des efforts cognitifs pour lire et comprendre le contenu de l'information.

Une visualisation de structures de l'information abstraites est requise. Il faut donc qu'elle soit appropriée, tant au niveau visuel, que de sa manipulation ainsi que de son exploration. L'utilisateur peut alors analyser les données affichées et interagir avec elles dans son environnement. L'usage d'un environnement immersif permet à l'utilisateur d'explorer des parties différentes de l'espace, affectées de niveau de détail, et de perspectives différentes.

Par exemple dans le cas d'une interface médiatrice à missions multiples, l'utilisateur peut changer facilement d'interface de commande suivant l'entité qu'il désire contrôler avec la possibilité d'afficher dans un coin de l'espace l'état sous forme d'informations synthétisées des autres entités du monde contrôlé. L'utilisateur est alors plongé dans un monde complètement interactif, métaphorique qui évolue avec le temps.

De plus, nous devons maintenir un certain équilibre entre la possibilité de voir et de présenter toute l'information. Il peut y avoir un effet d'encombrement visuel. Trop d'information, tue l'information. Quand la quantité d'informations dans une visualisation devient trop grande, l'utilisateur humain ne peut plus la percevoir dans sa globalité.

Il faut filtrer sans excès l'information. Ainsi, l'information relative à une entité mineure aura peut-être une durée de vie moins. La permanence des informations dépend naturellement du contexte et donc de l'intérêt de l'utilisateur.

Voilà, donc ce qu'il faudra prendre en compte pour pouvoir gérer les informations visuelles de notre interface. Maintenant, nous allons proposer une structure un peu plus proche de la réalité.

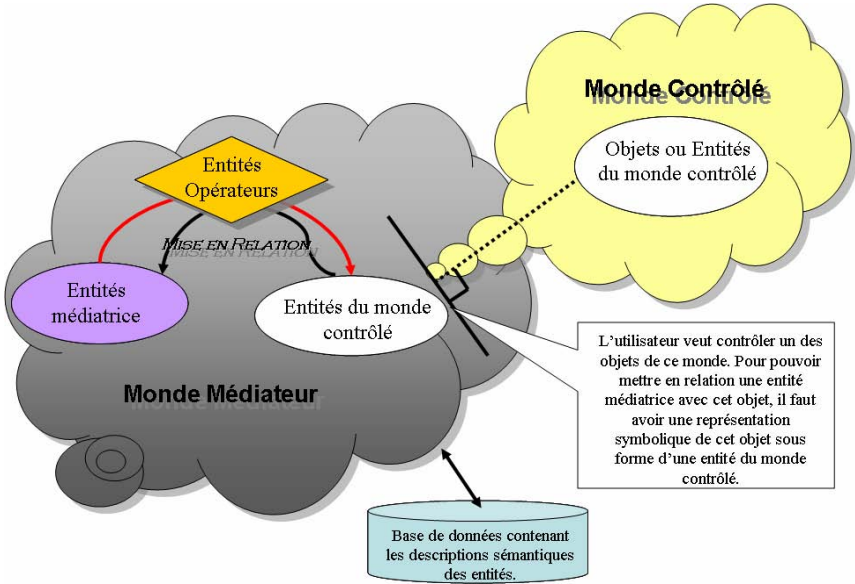


Figure 36 : Schéma simplifié du concept.

Le monde contrôlé contient les entités complexes que l'utilisateur désire contrôler et manipuler. Le monde médiateur, si on résume, c'est l'interface médiatrice contenant les entités médiatrices qui permettent de contrôler et d'agir sur les entités du monde contrôlé. Elle contient aussi des entités « opérateur » permettant de les mettre en relation ou d'agir directement sur l'entité du monde contrôlé. Nous avons pour cela distingué plusieurs types d'entité dans le monde médiateur. Chaque entité aura un rôle bien déterminé que nous allons aborder tout de suite.

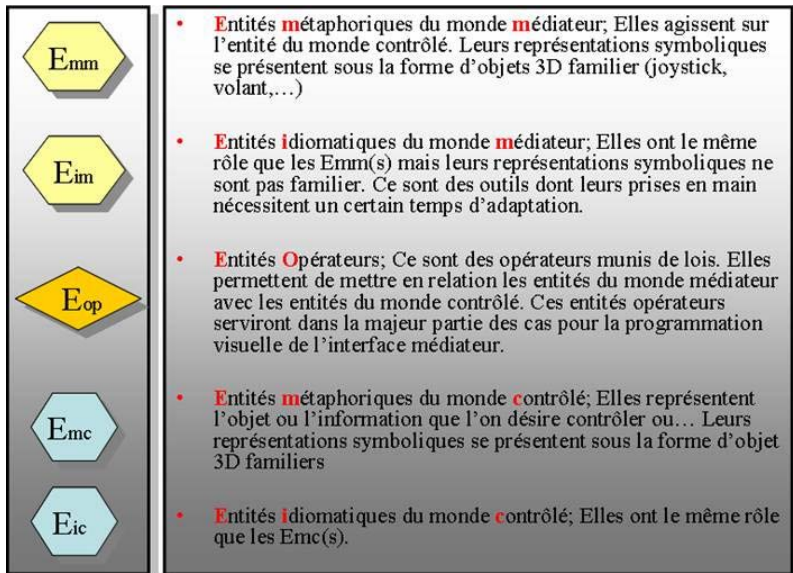
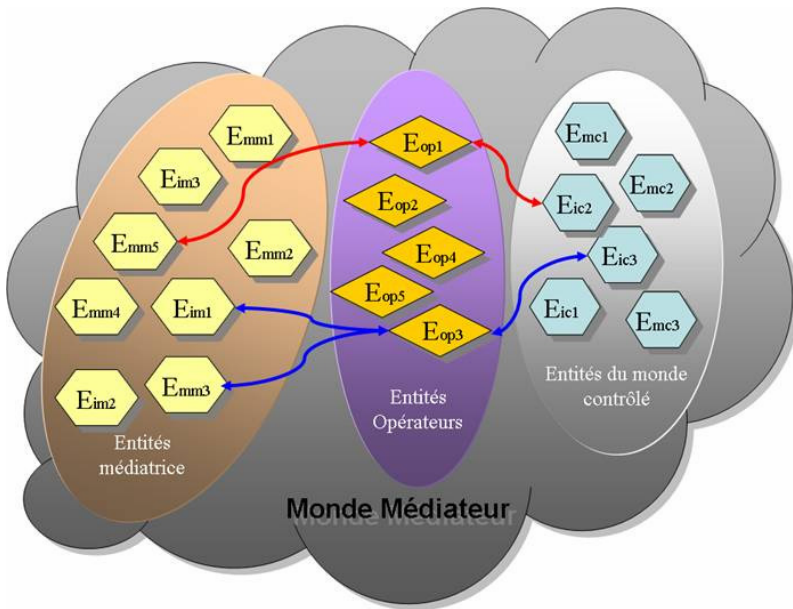


Figure 37 : Schéma plus détaillé du concept



Voilà donc ce que nous proposons. Concept qui peut être utilisé dans la création de futures interfaces médiatrices. Maintenant nous allons énumérer une suite d'idées qui peuvent être utilisées dans notre concept du médiateur.

Dans ce monde médiateur, l'utilisateur devrait pouvoir facilement prendre le contrôle d'une entité du monde contrôlé et obtenir rapidement l'interface de contrôle qu'il aurait préalablement définie ou qui lui serait proposée à des niveaux adaptés : novice, médium, expert. L'utilisateur pourrait disposer de son interface et lui ajouter, si nécessaire, de nouveaux outils de contrôle, définis par une description sémantique et accessibles dans une base de données.

L'utilisateur devrait pouvoir à tout moment modifier son environnement sans perturber le fonctionnement de l'interface médiatrice agissant sur les différentes entités du monde contrôlé. On devrait pouvoir aussi lui proposer une interface minimaliste associée à l'entité qu'il veut contrôler, ainsi que l'interface de contrôle global.

L'idée est donc d'accéder à une fiche descriptive des moyens de contrôle utilisé pour agir sur l'entité du monde contrôlé mais il faudrait aussi donner des limites car on ne peut pas faire n'importe quoi.

On devrait aussi restreindre au maximum l'information visible ou perceptible par l'utilisateur et donc regrouper les sources d'informations afin de présenter à l'utilisateur une vision synthétique des choses. Un filtrage de la visualisation de l'information devrait être intégré afin d'éviter toute surcharge dans le monde médiateur. Il ne faut présenter que les informations pertinentes et le strict minimum au niveau des moyens de contrôles.

L'interface médiatrice devrait pouvoir gérer les temps de latences de transmission des actions et de réactions. Nous devons avoir la possibilité de donner une tâche à exécuter à une entité du monde contrôlé. L'exécution de cette tâche et son accomplissement devrait être supervisés par notre système.

L'interaction homme machine devrait permettre à l'utilisateur d'accomplir des tâches de façon efficace, avec une bonne productivité, en toute sécurité, en prenant plaisir à le faire, en apprenant rapidement à utiliser le système.

Lors de la conception de l'interface, on devra prendre en compte la vitesse d'apprentissage de la personne, sa capacité de mémorisation, sa curiosité, son niveau de compétence dans la tâche. Un système de base de données devrait être donc créé pour enregistrer le maximum d'informations concernant la personne (Physiques, physiologiques, Socioculturels). Cette base de données serait accessible via Internet et fournirait les éléments nécessaires pour l'interface. Cette interface devrait détecter les tâches répétitives et proposer à l'utilisateur des solutions optimales.

Face à cette énumération d'idées, des questions restent encore en suspens à savoir:

Comment définir les caractéristiques et les propriétés de différentes entités présentes au sein du monde médiateur ? Pour cela nous pouvons effectuer pour chaque entité une description sémantique (sa forme visuelle, ses contraintes de manipulation, ses propriétés, son comportement, ses connecteurs). Il faut donc ajouter à cette structure une base de données fournissant les informations nécessaires.

Comment connecter sous contrainte les entités du monde médiateur et du monde contrôlé ? Simplement en comparant les connecteurs des différentes entités. Une sorte de clef soit numérique de la forme binaire 10101111 ou par mot clef.

Comment définir les entités opérateur permettant de mettre en relation ou d'agir sur les autres entités par la programmation visuelle ? Cet outil doit être constitué de lois définies dans le cas applicatifs par l'utilisation de pythons et de prologue. Ces opérateurs pour plus de clarté devront avoir des structures visuelles différentes, ce qui nécessitera la création d'une grammaire visuelle. La définition des opérateurs pourrait être : Les opérateurs de base : Création, sélection, activation, bouger, connexion, allez à, jouer, stop, voir, remplacer, effacer, oui, non ; Opérateurs de logique : and, not, or xor ; Opérateurs de logiques floues ; Opérateurs définis par l'utilisateur.

Comment allons-nous présenter les choses visuellement ? Une gestion de la répartition de l'information visuelle est nécessaire.

Comment connecter les périphériques hardwares avec le monde médiateur ? En utilisant le même moyen que ceux utilisé pour les entités par une description sémantique.

Finalement, nous devons répondre à toutes ces questions que pour le développement des futures d'interfaces médiatrices.

#### 4.4. DISCUSSION

Nous avons vu jusqu'ici que nous pouvons construire une multitude d'interface médiatrice basée sur la même idée. Mais ne serait-il pas intéressant que l'utilisateur en immersion totale puisse construire en temps réel sa propre interface à travers des méthodes de programmation visuelle. L'utilisateur pourrait grâce à cette programmation visuelle mettre en relation les entités du monde médiateur avec ceux du monde contrôlé. Il faut rappeler que la programmation visuelle est un outil permettant de spécifier un programme. Elle est définie communément comme l'usage d'expressions visuelles dans un processus de programmation. Une langue visuelle est un ensemble d'arrangements spatiaux de symboles (texte, graphiques, icônes, et ...) avec une interprétation sémantique qui est utilisée pour réaliser des actions. Ces éléments graphiques ou iconiques ou ... peuvent être alors combinés et manipulés par l'utilisateur suivant quelques règles ou « grammaire spatiale » spécifique à ses règles de construction.

La programmation visuelle est une manière simple, permettant par un jeu de constructions de concevoir sa propre interface et aussi de définir des nouveaux outils affectés de règles nouvelles. Pour pouvoir créer notre interface de programmation, nous pouvons nous inspirer de celles développées dans de célèbres logiciels commerciaux tel que Virtools [Virtools] et LabView [LabVIEW]. Le paradigme de la programmation visuelle a plusieurs avantages, quand il vient spécifier les interconnexions entre entités dans une forme de liens entre les ports d'entrée et de sortie [Ichikawa1987], [Vodslay1997]. La programmation visuelle requiert des symboles (icônes 3D, objet 3D), donc des primitives formes simples avec comme attributs secondaires couleur et texture de la surface. Nous pouvons proposer un langage visuel de programmation constitué de formes, une sorte de grammaire de formes.

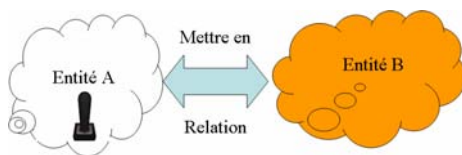


Figure 38 : Mise en relation des deux entités

L'objectif est de mettre en relation deux entités A et B. A peut être une entité médiatrice et B une entité du monde contrôlé. L'opérateur « Mettre en relation » peut être traduit sous forme visuelle. L'utilisateur sélectionne un des

opérateurs par exemple une tige bleue et relie l'entité A à B de manière manuelle. Bien entendu un mécanisme en arrière plan doit vérifier si cela est possible ou non.

Une description sémantique des éléments dans la scène est alors nécessaire afin de définir les règles permettant de mettre en relation et d'interagir avec les entités. Comme par exemple, nous pouvons définir l'opérateur de manipulation. Opérateur dont l'instrument ou la représentation symbolique est la main qui va agir directement sur l'objet. L'objet va donc changer d'état tout en respectant les contraintes qui lui ont été affectées. Si l'objet a le droit de bouger, de se déplacer, l'utilisateur verra son objet bouger.

Nous pouvons voir aussi notre interface médiatrice comme cela. L'utilisateur grâce à sa télécommande virtuelle : « *menu pétales* » dont nous verrons sa description un peu plus tard, sélectionner les entités médiatrices dont il a besoin et les positionner dans son espace virtuel de travail. L'utilisateur en immersion totale va alors connecter les entités médiatrices avec l'objet qu'il désire contrôler via une interface de programmation visuelle et en respectant les paramètres de connexions définies dans la description sémantique de chaque entité.

#### 4.5. CONCLUSION

Nous avons donc proposé jusqu'ici notre concept de médiateur. C'est-à-dire une interface médiatrice pour contrôler des entités d'un monde contrôlé réel ou virtuel avec pour contrainte d'obéir en temps réel aux ordres que nous envoyons via un canal de communication (TCP/IP ou UDP).

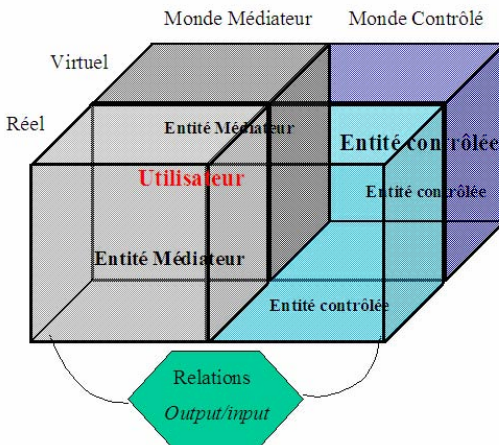
Autre remarque, la variété des modes d'interaction et d'interface multimodale conduit inévitablement à orienter les recherches vers une meilleure généricité du système. Ce qui implique que, pour palier les défauts majeurs, l'interface médiatrice doit être suffisamment flexible pour pouvoir connecter n'importe quels périphériques hardware. Un descripteur au sein de notre application est donc nécessaire pour pouvoir spécifier ses agissements sur l'interface. Par exemple le joystick ou la « *station haptique* » agit sur quoi, envoie quoi, et reçoit quoi comme données. La « *station haptique* » donne des données de spatialisation de la main de l'utilisateur qui sont utilisées pour le mouvement de la main virtuelle dans l'environnement médiateur.

Autre remarque concernant les interfaces, nous pouvons constater que nous copions dans la majorité des cas le réel pour le retranscrire dans le virtuel. D'où la question pourquoi ne pas créer d'autres outils interactifs qui n'existent pas dans notre monde réel ? Une des questions qu'il nous faut naturellement se poser : devons nous forcer à nous adapter à de nouvelles formes d' « affordances » ou est-ce que c'est au concepteur de trouver les meilleurs moyens pour optimiser notre capacité de prise en main. Il est clair que dans la définition d'outils de l'interface, il faut de la cohérence et de la compréhension qui passe par un modèle conceptuel explicite et perceptible. Un bon concepteur ne néglige aucune donnée et doit tenir compte du comportement et de la réaction commune des personnes vis-à-vis d'un outil ou d'une tâche à effectuer.

---

## CHAPITRE 4 : APPLICATION DU MEDiateUR

L'objectif de ce chapitre est de vous présenter quelques applications qui utilisent le concept du médiateur c'est-à-dire des moyens de commandes ou entités médiatrices pour contrôler des entités complexes. Les résultats de nos expériences testées par des candidats, nous ont fourni de précieuses informations sur différents points déjà évoqués et nous ont permis d'échafauder les bases d'une interface médiatrice idéale.



La figure ci-contre représente, un résumé sommaire du concept du médiateur pour tous les cas possibles de configuration. L'utilisateur interagit soit dans un environnement virtuel ou réel avec plus ou moins de degré d'immersion sur un monde contrôlé. Suivant l'application développée, des relations ont donc été créées entre le monde médiateur et le monde contrôlé.

Nous allons maintenant aborder point par point les différentes applications qui ont été développées et qui ont permis de tester notre concept. Nous verrons en profondeur la description de chacune des applications, les difficultés rencontrées ainsi que les enseignements à en tirer.

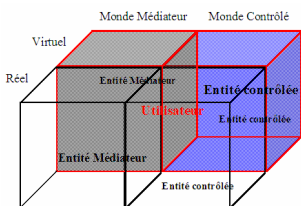
## 5.1. LE MEDIEUR DANS UN CONTEXTE DE MENUS 3D

Nos premiers travaux ont débuté par l'étude des différentes techniques de manipulation directe sur des objets 3D dits complexes et non convexes. Etant donné que les calculs de collision en temps réel prennent du temps, nous devons trouver des compromis. Les résultats obtenus étaient loin de nos espérances et c'est pourquoi nous nous sommes dirigés peu à peu vers une interaction indirecte via les menus 3D. Notre première étude visait d'une part à faire un état de l'art sur les menus 3D et d'autre part de comprendre les difficultés d'utiliser ce moyen de communication. L'objectif était de pouvoir manipuler n'importe quels objets via une interface de type menu et aussi déterminer le type d'immersion auquel le sujet devait être assujéti (non immersif, semi immersif, immersif).

### 5.1.1. Objectifs

Les objectifs étaient de :

- Tester les concepts des menus 3D.
- Observer le comportement des utilisateurs novices ou experts dans nos environnements virtuels que nous avons développés



#### Remarques par rapport à l'interface médiatrice :

Le monde contrôlé se trouve dans le monde médiateur. Nous interagissons de manière indirecte par l'intermédiaire de menus 3D sur des objets plus difficiles à manipuler

#### Remarques générales :

Des travaux restent à faire sur la sélection d'un objet par désignation et sur la compensation des problèmes hardware. Les résultats obtenus nous ont conforté sur l'utilisation du « *menu pétale* » dans notre interface médiateur. Outil tout le temps disponible sous la main..

**Article de référence :** Interaction Techniques : 3D Menus-based Paradigm [Lemoine2003]



Les menus 2D ont été traduits finalement dans un environnement 3D sous forme de plan. On a donc repris les mêmes techniques, c’est-à-dire des menus standards tels que les menus déroulants et les menus contextuels permettant aux utilisateurs de choisir facilement des opérations à effectuer. Ces menus peuvent être placés librement dans le monde virtuel. Ils peuvent être connectés à un objet virtuel ou être lié a une partie du corps de l'utilisateur.

L’interface permet d’adapter le contenu du menu à la position choisie et d’associer l’action choisie à l’objet se trouvant à cette même position. C’est pour cela que Namgyu Kim [Kim] a fait une grande étude sur l’ergonomie des menus 2D en environnement 3D (emplacement & direction de la vue, architecture de représentation du menu et ...). Nous connaissons ces formes et ainsi que leurs fonctionnements.

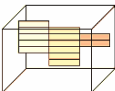
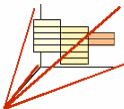
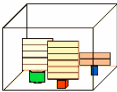
 <p>World Fixed</p>  <p>View Fixed</p>  <p>Object Fixed</p>	« Pull-down »	Un système de menu que nous connaissons très bien, ne montrant que les parties, les branches explorées dans les menus pendant la tâche de la sélection.
	« Pop-up »	Le menu disparaît une fois la sélection est faite.
	« Stack menu »	Un système du menu classique dont les étapes disparaissent une fois la sélection est faite. Principe du panier de basket.
	« Oblique/Layered »	C’est une représentation de menu affichée dans un mode oblique, où sa structure est organisée et affichée par des couches.

Figure 39 : Distribution spatiales des menus  
et architecture de menus classiques dans un environnement virtuel

D'autres chercheurs ont exploré d'autres voies, d'autres combinaisons par une interaction multimodale. Jacoby, par exemple a implémenté dans son application des menus 3D virtuels de type « pop-up » et « pull-down » et utilisé quatre types de gestes pour pouvoir sélectionner les items du menu. D'autres techniques ont été mises en œuvre pour la sélection qui s'opère dans les menus. Par exemple, nous pouvons sélectionner directement avec une main virtuelle, ou utiliser le système de reconnaissance vocale [Darken1994] ou utiliser des techniques d'occultation [Pierce1997]. Le plus difficile dans l'utilisation des menus 3D, est la sélection. La sélection directe par un outil de pointage non contraint dans l'espace 3D n'est pas optimale. L'utilisation d'un rayon virtuel contrôlé par la main comme outil de pointage atténue les difficultés en éliminant les contraintes de sélection en profondeur [Lindeman1999], mais il y a diminution des performances [Paljic2002]. La sélection par occultation est une technique interaction basée sur un concept 2D. La sélection est basée sur le rapport entre l'image projetée sur l'écran 2D du HMD de l'utilisateur. La sélection s'opère de la manière suivant, on calcule le vecteur directeur d'origine la position de la tête de l'utilisateur et la position de la main pointant l'objet occulté. Ce qui implique que nous devons avoir alors une bonne coordination entre la tête et la main. Nous pouvons utiliser aussi d'autres techniques, principe du faisceau laser, un système d'aide à la désignation de l'objet sélectionné, ayant pour origine la position de notre main dominante.

Quant à Rudy Darken [Darken1994], il utilisa le système de reconnaissance vocale pour la sélection des items dans les menus. Il suffisait simplement de lire les commandes. L'avantage de cette technique est que les contraintes de sélection sont moindres. L'inconvénient c'est qu'il y a parfois un problème de lisibilité du texte dans l'environnement virtuel, ce qui peut perturber l'utilisateur. C'est un des problèmes que nous avons rencontrés dans notre implémentation de menus 3D.

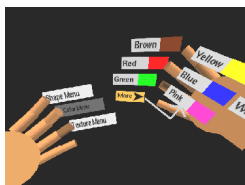
Une autre technique, utilise un traqueur sur la main et cinq gestes pour déterminer les intentions de l'utilisateur ou encore en utilisant une souris 3D avec cinq boutons actions.



Concernant le positionnement des menus ainsi que leurs distributions spatiales, les résultats des tests ont montré dans plusieurs recherches qu'une vue « World Fixed » serait une meilleure solution pour une compréhension de la structure générale des menus et de l'ordre et ainsi que des étapes dans la sélection des tâches spécifiques. D'autre part des recherches ont montré que si on n'utilisait

pas de reconnaissance vocale comme un moyen supplémentaire interactif, les méthodes gestuelles de sélection des tâches par un menu sont plus rapides et plus commodes comparées aux autres méthodes.

D'autres recherches ont été faites, faisant appel au sens proprioceptif avec une technique de placement des menus par rapport à des parties du corps. Les techniques ainsi développées sont tout de même intéressantes pour la manipulation directe des objets. Il permet d'avoir les objets centré sur son corps et ... [Mine1997].



D'autres métaphores ont été adaptées au monde 3D. Comme par exemple le concept TULIP présenté par Bowman [Bowman2001], qui est basé sur le pincement des doigts de la main. Les options de menu sont affichées comme des étiquettes rectangulaires au bout des doigts de la main virtuelle.

Ferneau [Ferneau1995] quant à lui, a introduit le concept de « 3D widget », c'est à dire des outils d'interactions comme des « slides », des objets outils intermédiaires agissant sur un objet sélectionné par l'utilisateur. C'est un concept intéressant que nous développerons un peu plus loin. Nous pouvons citer comme source d'inspiration pour une application 3D par exemple « Bryce de Corel Corporation » qui est logiciel de conception et de rendu de paysages 3D (générateur de terrains, des formes primitives et outils de modélisation). À la portée des non-spécialistes, son interface est intuitive, ludique, et esthétique, et suffisamment évocatrice.

Nous constatons que beaucoup de techniques basées sur nos interfaces de menus 2D ont été développées. Pourquoi ne pas retranscrire ce qui a été fait pour les interfaces 2D dans le monde 3D. D'autres recherches ont été faites sur l'ergonomie de menus ainsi que leurs formes. Des menus 2D linéaires ont été portés en 3D sous la forme de menus circulaires ou sous la forme de bandes. Comme par exemple les « Pie Menu » qui présentent :

des **avantages** : ils sont plus rapides et plus fiables que les menus linéaires [Callahan1988]. C'est un concept que l'utilisateur garde en mémoire car on utilise la capacité du corps à se souvenir du mouvement et de la direction, même quand l'esprit a oublié les étiquettes symboliques correspondantes disposées sur les zones d'intérêt. Autre élément

d'information, plus le curseur ou indice de position est loin de la zone, plus une sélection exacte est assurée.

des **inconvenients** : il faut gérer la disposition spatiale des menus et leurs profondeurs ainsi que la grandeur des étiquettes, c'est-à-dire de l'information. Un algorithme de disposition est requis qui devrait avoir trois buts : minimiser la dimension du menu, prévenir le chevauchement des étiquettes, et associer clairement des étiquettes avec leur direction. Une solution au problème des menus avec trop d'articles est de diviser les grands menus en plus petits, logiquement apparentés sous-menus. De plus on peut accéder rapidement aux différents niveaux puisque l'utilisateur se souvient parfaitement de l'itinéraire suivi pour accéder aux différents niveaux et à l'action désirée.

Certes le concept du « Pie Menu » a ses avantages et ses inconvenients, mais il ne tenait qu'à nous d'ajouter notre petite touche personnelle dans notre application. A partir des éléments observés, nous avons donc construit par la suite une sorte de « *menu pétales* » dotée d'icônes, d'un écran pour afficher de l'information, configurable au préalable par un programme externe générant son graphe. Nous verrons sa justification et sa description un peu plus loin.

### 5.1.2. Notre première implémentation de menus 3D

Nous présentons ici une interaction 3D en immersion totale en mode stéréovision.

Notre premier cas d'étude requiert l'utilisation de traqueurs magnétiques, d'un casque de réalité virtuelle (HMD) et de « *DataGloves* ».

Nous avons créé une image virtuelle de l'environnement de travail afin que les utilisateurs munis d'un HMD évitent les collisions avec le mobilier.

Les traqueurs utilisés fournissent les informations nécessaires pour positionner les mains virtuelles dans la scène et pour calculer la scène virtuelle observée. L'utilisateur utilise notre système de menus 3D de type « *Pull Down* » pour agir sur les objets qu'il désire contrôler. Les menus sont fixés dans l'espace.





Figure 40 : Interaction avec nos menus 3D. L'utilisateur sélectionne un bouton et confirme en faisant le poing avec la main gauche. Tâche demandant une certaine pratique.

Le but de l'expérience était d'évaluer, à l'aide de différents tests et d'un questionnaire les difficultés perçues par des utilisateurs experts ou novices, c'est-à-dire les erreurs faites dans sélection, le temps nécessaire pour accomplir une tâche, le soi-disant caractère expressif de la manipulation, l'ergonomie des menus, et des données subjectives sur le confort de l'utilisateur. Les tests consistaient à effectuer quelques tâches significatives.



Figure 41 : Visualisation des problèmes.

Les résultats observés, ainsi que ceux fournis par les tests avec d'autres utilisateurs nous ont montré que notre métaphore d'interaction était loin d'être efficace et rejoint les résultats observés par d'autres chercheurs. Beaucoup de problèmes ont été observés comme par exemple, réflexes intuitifs pour certaines personnes d'aller vers l'objet plutôt que d'utiliser les menus, l'estimation des distances entre main virtuelle et le menu. Ainsi que la gestion et l'organisation spatiale sont un problème (l'occlusion, positionnement des menus, texte illisible, fatigue musculaire et oculaire) ainsi que ceux dus aux matériels (bruit de traqueurs, câble).

Suite aux remarques constatées nous devons trouver un meilleur concept qui puisse être à la fois utilisé en mode « semi-immersif » (un utilisateur devant un grand écran) ou en immersion totale (utilisateur muni d'un HMD), disposant de menus 3D plus accessibles et plus conviviaux. Les questions que nous devons nous poser portaient sur la localisation et la forme des menus. Sa place au sein de l'espace virtuel est importante, car l'utilisateur doit pouvoir y accéder facilement. Il est possible est de placer des menus flottant sur les objets que l'on désire manipuler, mais d'après les résultats de notre enquête les utilisateurs préfèrent déplacer les menus et les avoir toujours à côté d'eux. On pourrait contraindre le menu à flotter toujours devant la vue de l'utilisateur, mais ce n'est pas une solution, car cela causerait des problèmes d'occlusions et de fatigues.

Finalement, l'idée retenue a été de réaliser une sorte de menu, fixé sur la main gauche de l'utilisateur, pour lui permettre de le déplacer comme il le souhaite, en le gardant toujours à portée pour pouvoir à chaque instant y jeter un coup d'œil.

La question est alors quel type de menu et sous quelle forme ? Jusqu'à maintenant dans un univers 3D, les menus classiques n'ont pas prouvé leur efficacité. Nous devons donc rechercher de nouvelles formes, de nouvelles structures et définir l'ergonomie de notre menu.

### 5.1.3. Recherche d'un nouveau design de menu 3D

Comme nous l'avons déjà évoqué auparavant de nouvelles formes de menus ont été étudiées jusqu'à maintenant ! Nous allons vous présenter ici quelques formes.

**Pie Menu :** Un « *Pie Menu* » [Newman1973] [Rollo1992] est une interface menu de forme circulaire surtout utilisé dans les environnements 2D permettant d'effectuer un ensemble de sélections de façon la plus naturelle possible. Le curseur de sélection placé au centre du « Pie Menu », permet de faire des sélections dans toutes des directions avec le minimum effort.

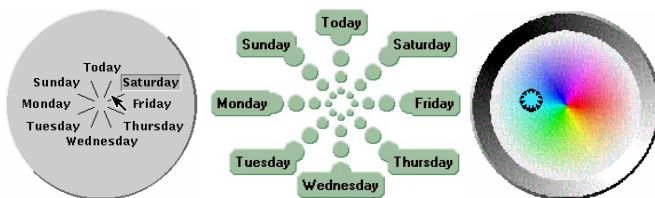


Figure 42 : Différents types de design avec lesquels on est plus ou familiarisé

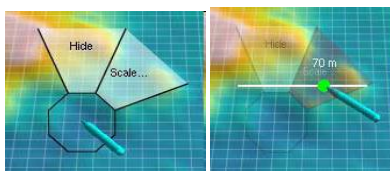


Figure 43 : Autre type In-“situ menu” et “slider” de control pour ajuster différents paramètres

La loi de Fitts explique les avantages de l'utilisation de ce type de menu, la vitesse de la sélection est plus rapide par rapport à un menu classique et le taux d'erreur est bas. Dès les années 50, Fitts a élaboré un modèle d'évaluation [Fitts] et détermina une équation permettant d'estimer a priori le temps pour se déplacer d'une cible  $C_i$  à une autre  $C_j$ . Le temps de recherche exigé avec un pointeur dépend de la cible de la région et de la distance.  $MT = a + b \log_2 (D_{ij} / W_j + 1)$ ; Où  $D_{ij}$  représente la distance entre les deux cibles,  $W_j$  la taille de la cible  $C_j$  et  $a$  et  $b$  sont des coefficients fixés de manière empirique. Elle a été utilisée pour estimer et évaluer la vitesse de saisie de texte des claviers par exemple. Mackenzie a estimé les coefficients :  $a=0.127$  et  $b = 1 / 4.9$  [Green2004]. Cette loi a été aussi utilisée pour estimer le temps nécessaire pour déplacer le pointeur souris par exemple vers une zone définie. Dans

le cas des « **Pie menus** », la loi de Fitt, d'après les travaux des chercheurs, prédit un bon temps en comparaison avec les menus traditionnels.

Ce type de menu est facile à utiliser pour les utilisateurs qui suivent les directions, les choix des items du menu car l'utilisateur n'a plus besoin de sélectionner une ligne dans une liste linéaire comme dans un menu contextuel standard mais peut tout simplement déplacer le pointeur dans la bonne direction et relâcher le bouton de la souris.

Les utilisateurs expérimentés peuvent agir plus rapidement et anticiper leurs actions, car ils connaissent le chemin. Les menus de type circulaire ont été développés dans les années 1960. L'université du Maryland a développé en 1986 et étudié l'ergonomie de ces types de menus et a cherché à les améliorer. En analysant par exemple la direction du mouvement avant la sélection, en désactivant lors de l'interaction certaines régions afin de contrôler et de minimiser les erreurs de sélection. La disposition circulaire permet une sélection rapide qui peut également être effectuée à l'aide d'autres périphériques tels que la souris, le « **TrackBalls** » ou un « **DataGlove** » par exemple. On peut alors adapter ces menus aux contraintes des appareils et adapter son apparence. Par exemple, le « **Pie Menu** » pourrait devenir une sphère à trois dimensions dans le cas où l'on utiliserait un « **DataGlove** ». Dans tous les cas, l'objectif est de fournir un outil de sélection basé sur une analyse gestuelle. C'est-à-dire un outil fiable d'utilisation pour les novices et les experts.

En conclusion, nous pouvons dire que les « **Pie menus** » sont faciles, rapides à utiliser, car ils fournissent un style gestuel d'interaction qui convient aux novices et aux experts.

Les « **Marking Menus** » sont semblables au « **Pie Menus** » et ont la caractéristique supplémentaire de reconnaissance gestuelle : l'utilisateur n'est pas obligé d'attendre que le menu soit affiché s'il connaît la direction de l'item qu'il veut sélectionner. [McGuffin2002], [Zhao2004]. L'évaluation des « **Marking Menus** », a montré des gains de temps d'un facteur 3,5. Kurtenbach et Buxton [Kurtenbach1993] ont étudié les limites et les performances de ces menus pour des experts. Et en 1999, Kurtenbach [Kurtenbach1999] proposa le « **Hotbox** », un nouveau système qui autorise la création d'au moins 1200 items de menus. Chaque item a un menu de type « **pull-down** » et chacun des cinq zones (milieu, haut, bas, gauche, droit), possèdent jusqu'à trois « **Marking Menus** ».



En conclusion, c'est une technique intéressante à utiliser surtout en deux dimensions mais compliquée à mettre en œuvre dans une dimension supplémentaire.

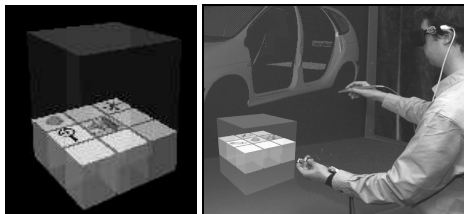


Figure 44 : Le C3 Menu

Puis, arrive enfin le C3 (Command and Control Cube), extension du concept de « *Marking Menus* » en 3D à l'aide d'une modélisation sous forme de cubes [Grojean2001]. C'est un menu évolutif qui s'adapte au niveau d'expérience de l'utilisateur. La sélection s'effectue avec la main non dominante afin de libérer la main dominante pour des tâches principales. Mais cela demande une certaine concentration que nous pouvons percevoir sur la figure ci-dessus.

#### « 3D Spin Menu » Interface :

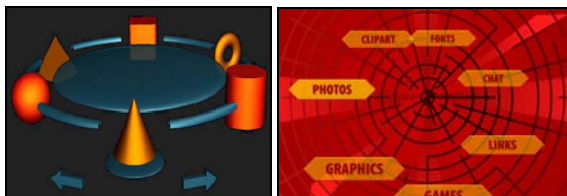


Figure 45 : 3D Spin Menu Interface

Ce menu tourne librement (comme un carrousel), et sa vitesse est modifiée selon la position du curseur de sélection par rapport au centre de rotation du menu. Dès qu'un des items du menu est en collision avec le curseur par exemple, la rotation du menu s'arrête. L'avantage de ce système réside dans le fait que ce n'est plus l'utilisateur qui va vers l'item du menu, mais l'item qui va vers l'utilisateur. L'inconvénient est le temps d'attente pour choisir 2 items successifs.

#### 5.1.4. Notre seconde implémentation : le « menu pétales »

Nous allons maintenant vous présenter notre seconde implémentation. Dans cet espace virtuel l'utilisateur reçoit de multiples informations (**visuelles** : texte sur les pétales, icônes, changement de couleur ; **sonores** : bruit de cliquage, synthétiseur vocale pour fournir plus d'informations ; **vibrotactile** : plus la main virtuelle est proche de l'objet, plus la fréquence de vibration de « cybertouch » augmente, *ceci était en option*). En utilisant le feedback visuel (changements de la couleur) nous exploitons notre capacité de détecter automatiquement les changements de formes, de couleurs, de textures des objets. Il fallait présenter l'information de façon intuitive et compacte.

D'autre part, pour interagir avec les menus l'utilisateur peut utiliser ses mains ou la parole. Pour choisir une commande par la parole, il doit prononcer des mots clefs qu'il peut lire sur un écran noir du menu. Un module de reconnaissance vocale qui sera dans un avenir proche un composant standard des interfaces traduira les mots pour exécuter les tâches. Nous ne devons donc pas renoncer à l'usage de commande vocale.

Ces différentes techniques mises en place permettent ainsi de s'adapter à la sensibilité de l'utilisateur par rapport à son environnement virtuel car nous ne sommes pas égaux par rapport à l'information que nous percevons.

#### Présentation de notre mini télécommande : le « menu pétales »

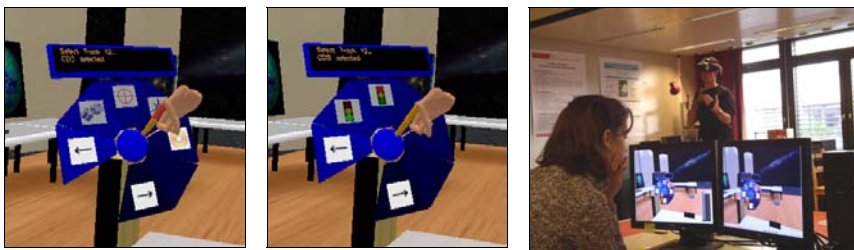


Figure 46 : navigation au sein du « menu pétales »

L'utilisation d'une combinaison d'interactions directes et indirectes avec utilisation de menus nécessite que ceux-ci ne doivent pas perturber l'utilisateur quand il manipule des objets de son univers 3D. Ainsi, la meilleure façon de faire les menus aussi discrets que possible et de permettre à l'utilisateur de les déplacer hors de son champ visuel était de créer une sorte de menu palette, fixé sur la main de l'utilisateur afin de lui permettre de le déplacer comme il le souhaite, tout en le gardant à portée de main. En gardant le « *menu pétales* » dans la main gauche, l'utilisateur a une meilleure vision de son environnement de travail. De plus, la grandeur du « *menu pétales* » est adaptable en fonction de l'encombrement visuel et de la dextérité de l'utilisateur.

Ce « *menu pétales* » est constitué :

- De pétales qui correspondent aux différents items du menu, qui sont différenciés par des icônes, facilitant ainsi l'interprétation et la reconnaissance visuelle.
- D'un petit panneau affichant des directives pour l'utilisateur ou de l'information concernant la confirmation de ses actions.
- D'un ellipsoïde situé au milieu, outil permettant de remonter dans l'arbre du menu, et donc d'un niveau.
- De deux items statiques permettant de naviguer à l'aide de ces flèches dans les items d'un sous menu.
- D'un dispositif sonore.
- D'opérateurs géométriques de bases (sélectionner, positionner selon des contraintes, orienter selon des contraintes et ...) symbolisés sur le menu par des icônes pour pouvoir agir de manière indirecte sur les objets positionnés dans la scène.

Désormais nous disposons, pour notre environnement, d'une sorte de mini télécommande qui n'affiche que le strict minimum pour ne pas accroître la surcharge cognitive de l'utilisateur. C'est un moyen interactif simple dont l'utilisation intuitive est facilitée par le recours à la métaphore de la main qui chauffe. Quand une des parties de la main est en collision avec l'objet, une couleur jaune dans notre cas avertit l'utilisateur et une information sonore qui peut varier en fonction de l'endroit pressé, confirme la sélection. Les problèmes d'estimation des distances que nous avons rencontrés dans les précédentes applications sont ainsi réglés.

D'autre part, nous calculons à chaque instant le point de projection entre la main et le « **menu pétale** » pour que le message affiché sur l'écran du menu soit en avec l'item (pétale) du menu. De plus, nous pouvons ajouter un dispositif à retour d'effort pour que l'utilisateur ressente la collision de sa main sur l'objet. Il sera utilisé ultérieurement, mais nous allons évoquer ici ses possibilités.

Avec un *retour d'effort passif* (principe utilisé par Bowman décrit avant), l'utilisateur tient, par exemple dans sa main gauche, une palette réelle ayant la forme de l'« objet pétale ». Le seul problème que nous pouvons rencontrer en utilisant ce dispositif est qu'un travail de calibrage est nécessaire pour faire correspondre la position de l'objet réel avec sa représentation virtuelle. L'utilisation de systèmes à traqueur optique est à exclure car il y a des problèmes d'occultation, il faut donc se servir d'autres systèmes limitant la perte et le bruit de l'information donnant la position de la main de l'utilisateur.

Avec un *retour d'effort actif*, il suffit d'utiliser la « **station haptique** » et un moteur de collision car l'exosquelette donne des indications, sur la position et l'orientation de la main de l'utilisateur dans le repère réel de l'appareil, qu'il suffit ensuite retranscrire dans notre environnement virtuel. L'utilisateur peut sentir alors « tactilement » le contact de la sélection sur l'« objet pétale ».

Par conséquent cette télécommande virtuelle : le « **menu pétale** », résout le problème de l'interaction dans l'immersif en fournissant à l'utilisateur une « méta interface » flexible et hautement configurable car les informations sur son arborescence sont stockées dans un fichier.

### Présentation du reste de l'application

Après avoir implémenté notre paradigme des « **menus pétale** », d'autres techniques d'interactions ont été ajoutées pour parfaire le fonctionnement de l'application :

- la sélection d'un objet en utilisant la technique de pointage de la « **bounding box** » englobant l'objet dit complexe
- les techniques de rotations et de translations à distance d'un objet sélectionné couplé au déplacement et à la rotation de la main
- ainsi que les techniques de navigation au sein de la scène.

La navigation est une fonctionnalité importante, car elle permet à l'utilisateur de se déplacer dans son univers virtuel. Si l'utilisateur est équipé d'un HMD, de « *DataGlove* », et de capteurs « *wireless* », il peut aller vers la scène naturellement ou désigner une direction avec sa main tout en restant immobile dans le monde réel.

Dans le cas semi immersif, en remplaçant le HMD par un grand écran, l'utilisateur n'a pas le choix, il doit utiliser des commandes gestuelles pour se déplacer : pointage de la direction de déplacement et utilisation du principe du tourniquet pour contourner la scène. Nous avons donc implémenté tous ces outils avant de réaliser des tests d'évaluation de notre interface.

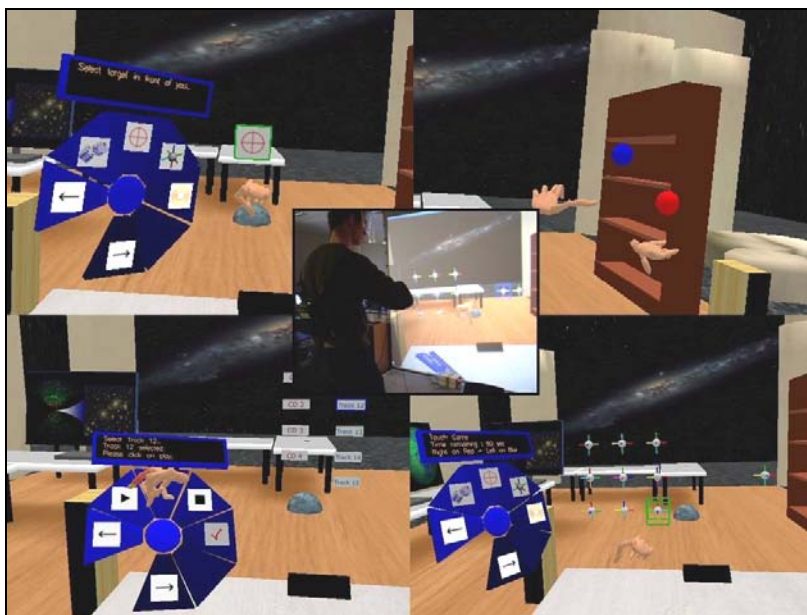


Figure 47 : Présentation des tests accomplis par différents utilisateurs dans notre monde virtuel.

Avant de réaliser les différents tests, il a été nécessaire de faire de nombreux réglages afin de ne pas perturber le candidat cobaye (fatigue et déconcentration risquant de biaiser les résultats). Il fallait donc calibrer les capteurs de positionnement (« *MotionStar* ») et les gants (« *DataGlove* ») pour la reconnaissance gestuelle.

Concernant le calibrage des gants, chaque candidat devait exécuter avec ses mains des postures bien déterminées permettant d'une part de sélectionner et d'orienter les objets et d'autre part à la navigation dans la salle.

Pour pouvoir évaluer nos concepts, nous avons développé quatre tâches spécifiques, effectuées en mode semi immersif et en immersion totale avec deux objectifs principaux : Mettre à l'aise l'utilisateur, le faire bouger au sein de cet espace, lui faire apprendre rapidement les techniques d'interaction implémentées, le faire agir et évoluer de manière naturelle mais également observer le comportement intuitif et l'adaptabilité à notre interface.

- Tâche 1 : Chercher et désigner des cibles, puis confirmer la sélection avec notre menu.
- Tâche 2 : Toucher les balles rouges et bleues apparaissant dans notre scène
- Tâche 3 : Repérage d'un objet commun dans la scène et a le placer sur l'objet témoin.
- Tâche 4 : Lire des directives formulées sur le panneau de contrôle du «*menu pétales*», et sélectionner des étiquettes affectées à une musique.

Les données des expériences ont été sauvegardées et analysées par la suite, et un questionnaire a été établi pour recueillir l'opinion des utilisateurs.

Les résultats ont montré que l'interaction avec notre «*menu pétales*» donnait de bons résultats car très peu d'erreurs ont été commises. Le problème se situait plutôt lors de la sélection des étiquettes, car étant plus petites, il était naturellement plus difficile de les désigner par la main. Ces problèmes de sélection sont de l'ordre de 40% sur grand écran et de 24% avec le HMD. Cette différence s'explique le comportement de l'individu qui n'a pas les mêmes réflexes devant un écran que sous un HMD.

Sous un HMD quand un objet est lointain, nous allons vers lui, alors que devant un grand écran, nous n'avons pas le réflexe d'utiliser l'outil de navigation pour se déplacer et se

rapprocher. Par analogie, nous éprouverions les mêmes difficultés si nous voulions appuyer la touche d'un clavier avec l'exterminé d'un manche à balais.

Il reste donc un travail à faire au niveau de la sélection des objets lointains et petits d'une scène virtuelle. Les utilisateurs ont trouvé que l'illusion était bonne ainsi que l'immersion dans cet environnement. La prise en main de l'interface était simple et facilement compréhensible car les tâches s'exécutaient, pour eux, de manière naturelle. Les utilisateurs préférèrent la navigation sous HMD bien que le champ de vision de l'appareil soit restreint et qu'elle fatigue la personne (physique et visuelle). Les résultats ont montré qu'en fonction de la tâche, on obtenait de meilleurs résultats en milieu « semi-immersif ».

Le dépouillement du questionnaire nous a révélé que les deux premières tâches ont facilité la prise en main de l'interface par l'utilisateur, que la manipulation n'était pas trop compliquée et qu'ils ont très bien anticipé les conséquences de leurs actions, que leur mémoire à court terme n'a pas été saturée par un excès d'informations et que l'immersion était satisfaisante. Après utilisation de notre «*menu pétales*», les personnes se disent satisfaites et jugent ce concept meilleur que celui des menus classiques moins intuitifs. L'utilisation quotidienne du téléphone portable est peut être une raison qui explique cela, nous y sommes habitués.

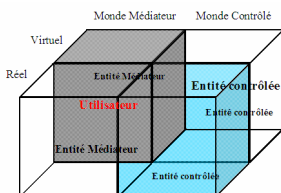
## LE MEDiateUR ET LE CONTROLE D'UN ROBOT

### 5.1.5. Objectifs

Les objectifs étaient de :

- Contrôler d'un robot mobile réel au moyen d'un tapis roulant et d'une analyse gestuelle.
- Diriger un robot mobile réel en "poussant" sa représentation virtuelle affichée sur un grand écran de projection.
- D'utiliser comme moyens de contrôle, un tapis roulant pour le déplacement local de l'utilisateur et de « Webcams » pour une analyse gestuelle des mouvements de l'utilisateur.

**Remarques par rapport à l'interface médiatrice :**



Maintenant le monde contrôlé se trouve à l'extérieur du monde médiateur. Le robot réel appartient au monde contrôlé. L'utilisateur en « semi-immersion » contrôle le robot réel, en utilisant les informations provenant d'une Webcam, ainsi les informations (orientation, avant-arrière) traduites à l'aide d'une représentation virtuelle du robot. Un tapis roulant est utilisé comme moyen interactif, permettant de donner des ordres au robot réel.

**Article de référence:** The “Caddie Paradigm”: a free-Lo-motion Interface for Teleoperation. [Lemoine-1-2004]



### 5.1.6. Présentation de l'application

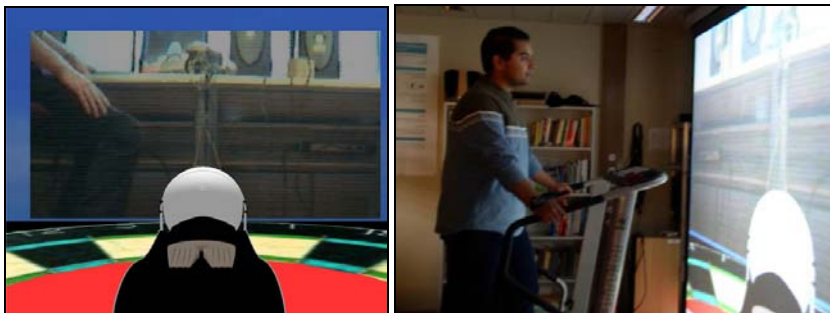


Figure 48 : Présentation du système pour contrôler un robot.

Représentation du robot sous la forme d'un avatar et de ce que voit le robot mobile réel ; Vue sur l'interface du tapis roulant

Nous vous présentons ici une interface de « téléopération » le « paradigme du caddie » basé sur le concept de médiateur permettant de contrôler un robot mobile réel à distance. Ce paradigme d'interaction permet de diriger en poussant un robot réel via une représentation virtuelle affichée sur un écran de projection. Ce système laisse l'opérateur marcher derrière ce robot virtuel en utilisant un tapis roulant.

Mais comment contrôler ce robot avec un système « téléopérateur » sachant que le champ de vision est restreint, la manœuvrabilité est limitée, la quantité d'informations perçues par l'opérateur sont très importantes ? Nous essayons de le manœuvrer avec des périphériques hardware : joysticks, volant,...et une caméra (une « spycam ») fournissant la vue du robot.

Après mûre réflexion, nous avons choisi un concept déjà utilisé dans plusieurs jeux vidéo : piloter un véhicule à la troisième personne, c'est à dire voir l'objet contrôlé par l'arrière afin de fournir à l'utilisateur des informations variées, affichables sur un écran.

Mais comment *pousser* le robot, comment le faire tourner sachant que l'utilisateur reste sur place ? Nous avons alors eu l'idée d'utiliser un tapis roulant unidirectionnel pour fixer un axe de déplacement. Mais comment donner l'ordre de tourner au robot, comment connaître la vitesse de déplacement ?



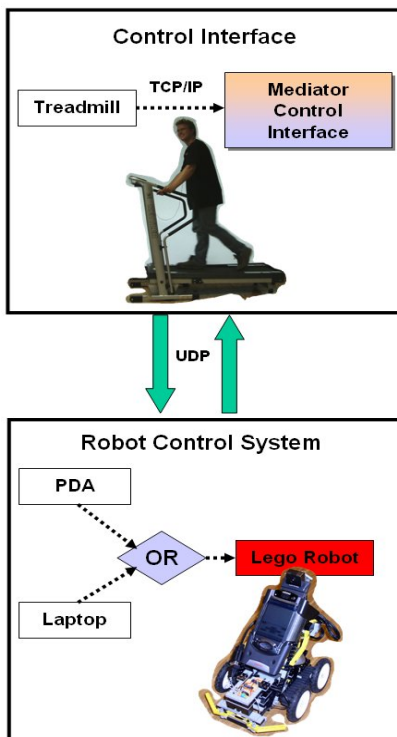
Figure 49 : vue sur les positions des membres supérieurs pour pouvoir contrôler la direction du robot mobile réel

Nous avons solutionné ce problème en ayant recours à deux « Webcams » ; L'une permet de déterminer la vitesse de déplacement en mesurant le temps écoulé entre les passages successifs d'une marque et d'analyser les mouvements gestuels de l'utilisateur à l'aide de zones prédéfinies dans les images filmées.

Quand les mains passent dans ces zones une information est donnée en fonction du protocole prédéfini :

- Pour changer la direction de parcours du robot (avancer ou rebrousser chemin) il faut placer simultanément chaque main dans sa zone d'action respective droite ou gauche.
- Pour tourner à gauche, on met la main gauche dans la zone prédéfinie à gauche de l'image
- Pour tourner à droite, on met la main droite dans la zone prédéfinie à droite de l'image.

L'utilisateur en semi-immersion voit sur un écran, l'image réelle du robot filmé par la « Webcam » et la représentation virtuelle de ce robot. Lorsqu'il donne au robot l'ordre de tourner, l'avatar ainsi que le robot tourne, confirmant l'exécution l'ordre donné. Voilà donc l'idée générale mais vous pouvez consulter



l'article pour avoir de plus de renseignements sur le robot utilisé et sur les connexions.

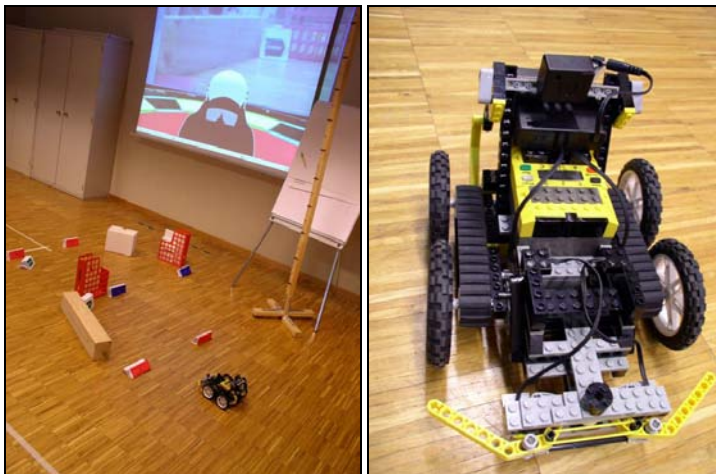


Figure 50 : Parcours d'obstacles pour un robot

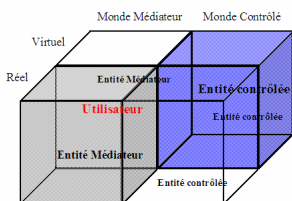
En guise de conclusion : marcher derrière un robot, est un moyen plus naturel que d'utiliser un « joystick », même si le niveau de précision dans les mouvements n'est pas comparable. Cependant, il est possible d'ajouter des moyens supplémentaires pour accroître la qualité de l'environnement et la précision des ordres transmis au robot car notre interface permet de gérer des informations visuelles telles que la cartographie du terrain avec localisation GPS, la proximité des obstacles, le niveau d'énergie,...

## 5.2. LE MEDiateur DANS UN CONTEXTE DE TELEREHABILITATION

### 5.2.1. Objectifs

Les objectifs étaient que :

- le patient évite de toucher les bords des tubes rouges, et suive le tuyau rouge dessiné par le thérapeute.
- le thérapeute avec son PDA comme interface dessine en temps réel les trajectoires et que les données des courbes soient envoyées par TCP/IP vers l'ordinateur qui commande l'application et la « *station haptique* »
- les données de chaque session soient enregistrées sous un format XML pour mesurer les performances du patient



#### Remarques par rapport à l'interface médiatrice :

Le monde contrôlé se trouve à l'extérieur du monde médiateur. L'utilisateur munit d'un PDA modifie les paramètres du monde contrôlé dans lequel le patient effectue des exercices.

#### Remarques générales :

Les exercices ont lieu en immersion totale, parce cela permet d'augmenter l'attention et la concentration du patient. Le fait d'être isolé du monde extérieur permet d'augmenter la notion de présence au sein de cet espace. (Représentation, sentir les retours d'efforts exercés par la « *station haptique* ». Mais garder le contact avec le patient est essentiel car un lien doit ce créer entre le thérapeute et son patient, car c'est un coach. A cet effet, l'image de la Webcam permet, à tout instant, de montrer et de corriger les gestes à accomplir par le patient.

**Article de référence :** Telerehabilitation: Controlling Haptic Virtual Environments through Handheld Interfaces. [Gutierrez-2-2004]

### 5.2.2. Présentation de l'application

Notre travail a été de développer une application dans le cadre de la télé-réhabilitation ayant pour finalité d'apporter, de manière ludique, une aide à des patients victimes d'un dysfonctionnement dans la coordination du mouvement des membres supérieurs. Cette application est basée sur l'usage d'une interface « *haptique* » et d'un environnement virtuel totalement configurable. Le patient, immergé dans un environnement virtuel (VE), muni d'un casque de réalité virtuelle (HMD) et assis dans une « *haptic workstation* » doit effectuer un certain nombre d'exercices. Dans un monde virtuel, le patient immergé se voit représenté par un avatar possédant des mains virtuelles.

Il doit essayer de suivre des lignes brisées ou des courbes à l'aide de ses mains comme s'il les dessinait.

- ✓ La « *station haptique* » permet de contraindre la personne à suivre la bonne trajectoire pour réussir l'exercice.
- ✓ Un thérapeute utilisant un PDA peut reconfigurer à distance l'environnement virtuel du patient.
- ✓ Une Webcam filme en permanence le médecin et l'image de celui-ci est affichée dans l'environnement du patient qui peut ainsi garder un contact permanent avec lui.

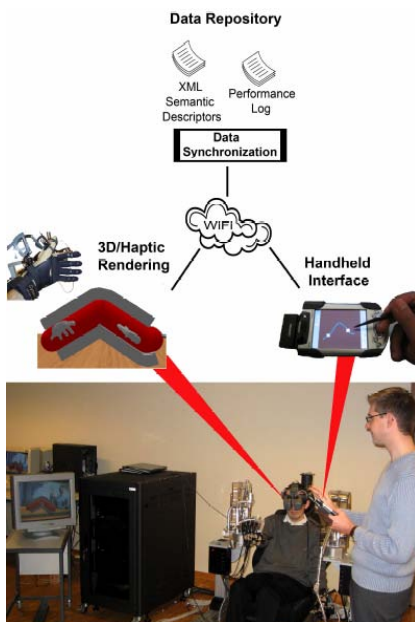


Figure 51 : Schématisation de l'application



Figure 52 : Le thérapeute donne des directives au patient.

Un psychiatre a jeté un coup d'œil sur notre système et a trouvé que cette idée d'intégrer cette fenêtre de contact était une amélioration très positive pour les environnements de la thérapie virtuelle, car elle permet au patient de rester en contact avec le monde réel. C'est une aide précieuse qui présente comme autre avantage incontestable le fait que le thérapeute n'a pas besoin d'être sur place pour diriger, dialoguer avec le patient lors des exercices thérapeutiques. De plus, pour chaque session, les données des mouvements du patient sont sauvegardées dans un fichier permettant d'analyser ultérieurement les performances du patient et de mesurer le progrès accompli au cours des exercices

### 5.3. LE MEDIEUR DANS UN CONTEXTE DE TELEOPERATION

#### 5.3.1. Objectifs

Nous abordons ici des applications tests qui sont un peu plus proches de notre concept de médiateur. Nous avons choisi de combiner la manipulation directe avec une manipulation indirecte. Le concept est assez simple. L'utilisateur agit indirectement sur le monde contrôlé par l'intermédiaire d'outils du monde médiateur. Dans ce monde médiateur, l'information provenant de l'objet du monde contrôlé est traduite sous la forme de stimuli visuels, acoustiques, ou retour d'effort. Il est ainsi possible de contrôler des objets lointains via une sorte de télécommande virtuelle. Pour illustrer ce concept et la faisabilité de notre système, nous avons décidé d'implémenter cette petite application : conduire une voiture de manière indirecte

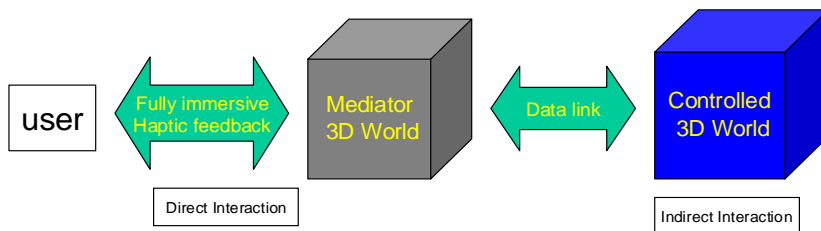
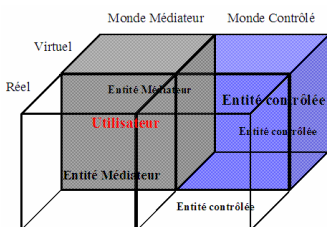


Figure 53 : Schéma de principe

Les objectifs étaient de :

- Développer des applications focalisées dans un contexte de téléopération.
- Etudier la faisabilité et l'intérêt du concept.

Remarques par rapport à l'interface médiatrice :



Le monde contrôlé se trouve à l'extérieur du monde médiateur. L'utilisateur est en immersion totale. Les périphériques de contrôle sont maintenant virtuels et modélisables même si nous utilisons une « *station haptique* » pour faciliter l'interaction et donner une certaine rigidité aux médiateurs. L'utilisateur a ainsi la possibilité de construire sa propre interface médiatrice.

**Remarques générales :**

Outre les problèmes au niveau « *haptique* », déjà évoqués dans un long chapitre, un certain temps d'adaptation est nécessaire pour pouvoir exécuter les tâches de façon optimale. Mais une étude plus poussée, de la répartition des informations, de la gestion judicieuse de la mémoire à court terme des individus et de l'utilisation des moyens mnémotechniques, permettra sans nul doute faire des progrès dans la rapidité d'exécution.

**Article de référence :**

Mediators: Virtual Interfaces with Haptic Feedback [Lemoine-2-2004]

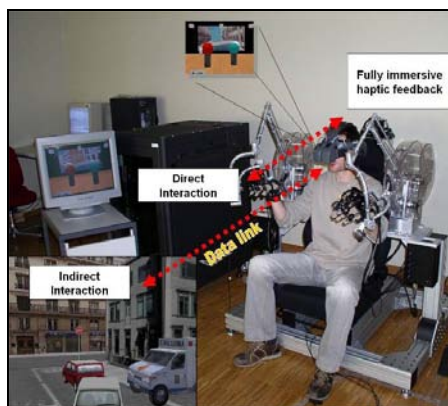
**5.3.2. Présentation de l'application**

Figure 54 : « *haptic workstation* » :

Située dans le contexte de la téléopération, l'application que nous avons développée, nous a permis de tester la faisabilité et les problèmes dans le contexte du médiateur. Nous avons donc ainsi développé un prototype de médiateur 3D, constitué d'un casque de réalité virtuelle (HMD) et d'une « *station haptique* » exerçant des retours d'efforts sur les bras de l'utilisateur. Le système permet de manipuler des objets virtuels appelés médiateurs servant d'interface virtuelle pour conduire une voiture située dans un environnement plus complexe et distant appelé monde contrôlé.



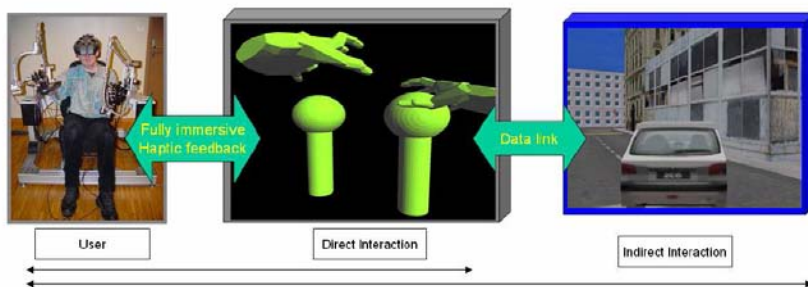


Figure 55 : Illustration de l'application étudiant

Notre étude consistait à étudier les réactions de l'utilisateur lors de la conduite d'une voiture dans une ville à l'aide de périphériques virtuel (joystick dans ce cas). L'utilisation de tels moyens de contrôle était plutôt déroutante car le dispositif de pilotage n'était pas le même que celui d'une vraie voiture. La manette de droite permettait d'avancer et de reculer et la manette de gauche permettait de tourner à droite ou à gauche.

Cette façon de conduire n'était pas évidente, mais avec de l'entraînement, on y est arrivé. Nous avons par la suite remplacé une des manettes par un volant virtuel plus maniable pour permettre une conduite plus proche de la réalité

Pour pouvoir utiliser ces périphériques correctement et de manière plus aisée, un stimulus de retour de force au niveau des bras et doigts a été mis en place via l'utilisation de la « *station haptique* ».

Afin de minimiser les problèmes dérivés des algorithmes de calcul de collisions, on ne peut manipuler que des objets simples. L'introduction d'objets complexes et non convexes augmente les temps de calcul et les forces de réaction sont moins fluides et provoquent une gêne pour l'utilisateur.

Nous avons défini un catalogue d'objets disponibles, simples et interactifs. Le prototype que nous avons ainsi développé, nous a conforté dans la faisabilité du concept du médiateur.

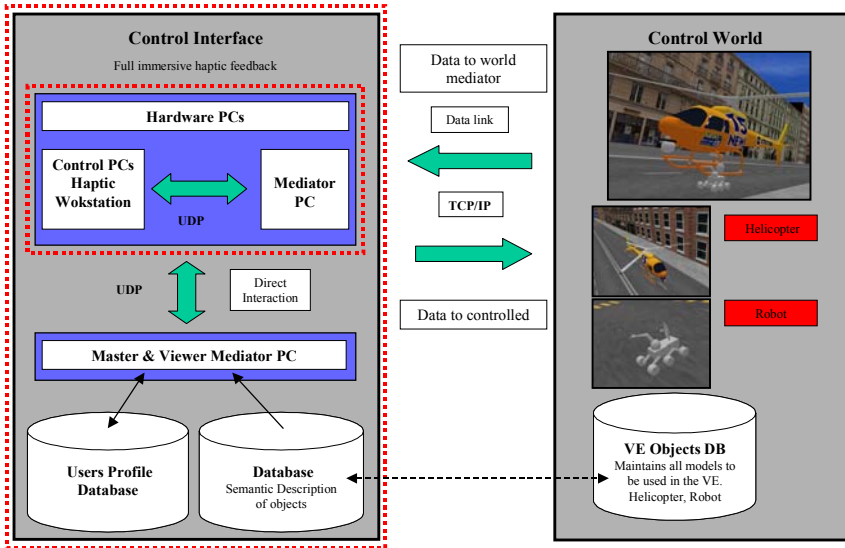


Figure 56 : Illustration du système 2

Confortés par les résultats obtenus, nous avons décidé d'ajouter de nouvelles fonctionnalités que nous allons vous présenter dans notre second développement.

Nous sommes toujours dans le contexte de celui de la téléopération mais l'interface de contrôle est entièrement paramétrable ce qui n'était pas le cas dans notre application précédente, entièrement monolithique et figée donc non configurable. La flexibilité étant de rigueur, nous avons doté notre interface médiateur d'une ontologie basée sur une description sémantique des objets, des accessoires qui la compose et des entités que nous voulons contrôler (exemple un robot, un hélicoptère dans notre cas).

Une « *station haptique* » avec retour d'efforts et un casque de réalité virtuelle (HMD) sont encore utilisés pour manipuler des objets virtuels (médiateurs) pour entités. L'interaction entre l'utilisateur et le monde médiateur est donc faite en utilisant des instruments virtuels tels que des « joysticks », « sliders », boutons, volants. Notre interface offre de grandes possibilités car nous pouvons créer n'importe quel outil médiateur sous réserve que sa représentation géométrique soit simple afin de ne pas perdre ce que nous avons gagné.

Les champs d'applications sont grands, puisque nous pouvons contrôler simultanément beaucoup de types d'entités (périphériques virtuels, entités du monde contrôlé. De plus l'utilisateur peut construire intégralement sa propre interface de contrôle, avec des outils de programmation visuelle rendus accessibles par le « *menu pétales* ». En effet, il dispose d'opérateurs interactifs de base ou périphériques virtuels de base (joystick, « slider », bouton,...) dont les fonctionnements sont précisés dans la base de données du descripteur sémantique évoqué dans le chapitre précédent.

Ce sont des objets médiateurs qui serviront comme interface pour pouvoir manipuler des entités du monde contrôlé. Ils sont dotés de mécanismes permettant de connecter un périphérique à l'objet que l'utilisateur désire manipuler.

Le « *menu pétales* » permet de changer le mode du système. Si nous nous trouvons dans une interface de programmation virtuelle, il est possible d'afficher, l'objet à contrôler avec ses inputs/outputs et la liste sélectionnée par l'utilisateur ou proposer des périphériques virtuels avec ses inputs/outputs pour pouvoir les connecter avant utilisation effective. Si nous nous trouvons dans le mode opératoire, nous pouvons mettre en scène les divers composants choisis par l'utilisateur (les médiateurs ou d'autres systèmes d'information) puis exécuter la simulation.

Dans le cadre de la « téléopération », l'utilisateur pourra contrôler son robot grâce aux manettes de commande virtuelles et visualiser dans son environnement ce que voit le robot sous différents angles. L'utilisateur dispose donc d'un système personnalisable et totalement configurable. Cette interface permet aussi de changer rapidement les périphériques de contrôle en fonction de la nature de l'objet du monde contrôlé de sorte que l'utilisateur puisse sélectionner un objet du monde contrôlé (robot hélicoptère) puis lui associer l'entité avec laquelle il va interagir. Rappelons que le médiateur peut être une interface multi mission.

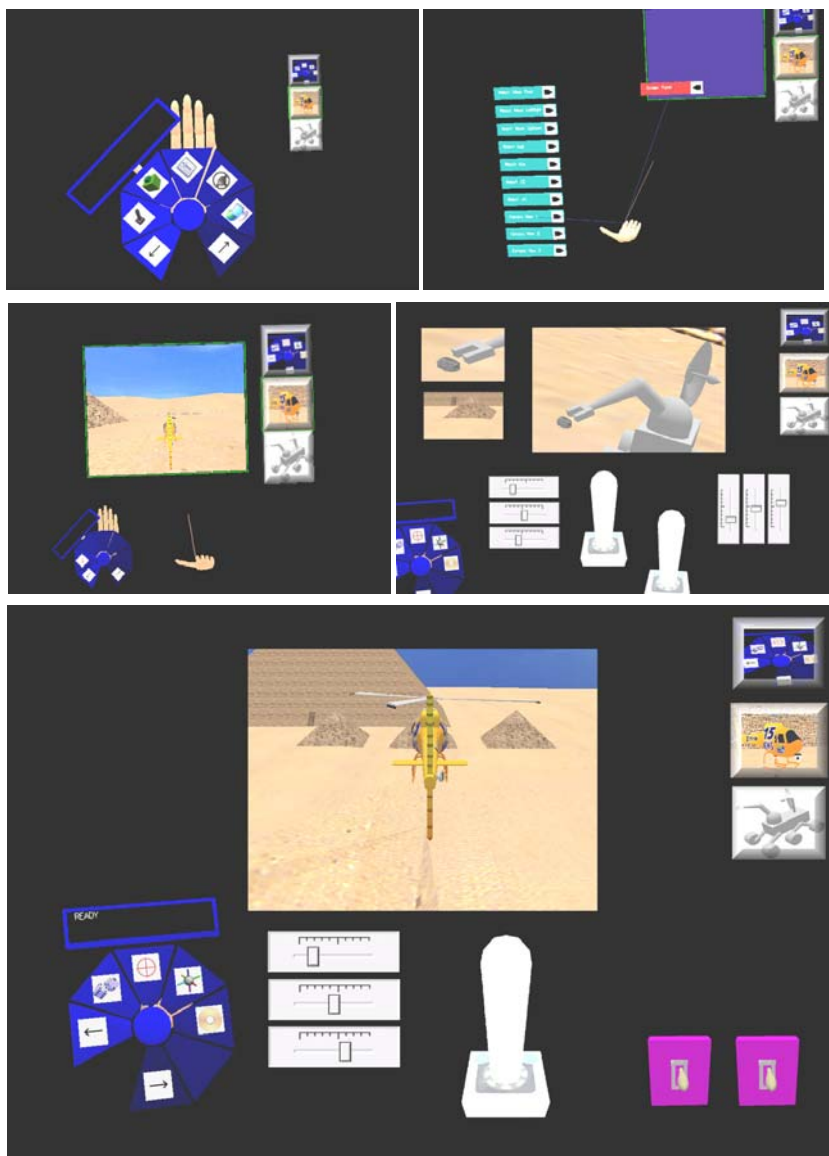


Figure 57 : Les étapes de la construction de l'interface de commande permettant soit de contrôler un hélicoptère ou un robot.

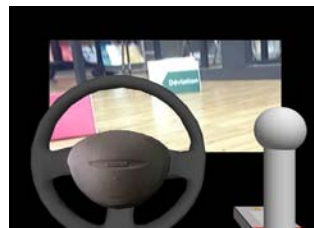
Cette conceptualisation offre une grande souplesse au niveau de l'interface et une certaine généralité. Sa mise en œuvre montre les possibilités du médiateur dans les scénarios « téléopérateur » « multi-mission ». Cette application peut être utilisée dans le cadre de simulateurs pour l'entraînement des utilisateurs. L'intégration de la programmation visuelle, ainsi qu'une description sémantique des entités dans l'espace médiateur simplifient énormément les choses en introduisant plus de souplesse, plus de dynamisme, plus fonctionnalité et offre la possibilité de tester des interfaces avant de les fabriquer, afin de réduire les coûts.

#### 5.4. INTERFACE MEDIATRICE POUR TELECOMMANDER UN ROBOT

Dans cette partie, nous allons raisonner dans un contexte de « téléopération » et voir comment le concept du médiateur peut y être appliqué. Tout d'abord, nous allons définir notre environnement de travail qui nous semble le plus approprié parce que l'utilisateur sera totalement plongé dans un monde immersif avec utilisation d'une « *station haptique* » à retour d'effort comme moyen interactif. L'utilisateur agit à l'aide de ses mains directement sur les éléments du domaine, et ces objets abstraits, devenus consistants grâce au retour d'effort, se comportent comme les objets réels. L'interface devient elle-même un monde explicitement représenté où aucun intermédiaire n'existe entre l'utilisateur et le domaine ainsi modélisé. Malgré les problèmes déjà évoqués auparavant dans le chapitre de l'« *haptique* », l'utilisateur peut manipuler n'importe quel objet virtuel dans son environnement virtuel en mettant par exemple les manettes de commandes dans le monde virtuel. Ces manettes virtuelles, que l'on peut placer où l'on veut, sont alors des instruments médiateurs permettant de contrôler une entité du monde contrôlé.

On pourrait ainsi piloter un robot par une interface médiatrice de plusieurs façons :

- Utiliser des manettes de commande, style joystick virtuel, volant virtuel. Nous utilisons ce système de commandes dans la vie courante mais il serait transposé dans le virtuel avec un système à retour d'effort.



- Utiliser des commandes gestuelles, c'est-à-dire qu'au niveau de l'interface, l'opérateur voit sa main et les indicateurs visuels de positionnement de sa main virtuelle. Le principe de commande est simple quand l'opérateur avance la main, le robot avance. Pour tourner à droite, il positionne sa main à droite et pour tourner à gauche, il positionne sa main à gauche. La main virtuelle de l'utilisateur est soumise à un système de force qui la repositionne au point mort afin que le robot ne bouge plus. On peut même envisager le cas où le robot s'approche d'un obstacle. Des capteurs estiment la distance et informent le système de l'imminence de la collision. Le système contraint, via la « *station haptique* », à positionner la main de l'utilisateur en arrière. Comme cela, il est informé physiquement et aussi par des alertes sonores ou virtuelles.
- Utiliser des commandes vocales du style « avance », « recule », « gauche », « droite », « stop », donc des messages courts, simples, contextuels qui requièrent une réponse immédiate lorsque la transmission ne peut se faire par le canal visuel C'est efficace si l'utilisateur n'a pas à prononcer ces mots trop fréquemment. L'utilisation de la reconnaissance de la parole est nécessaire dans les cas où l'interactivité est contrainte.

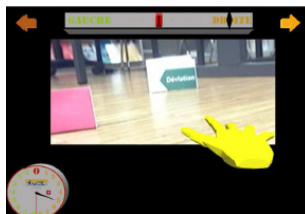


Figure 58 : Présentation des périphériques de contrôles virtuels pour contrôler un robot

Et si le robot était un bras manipulateur, on pourrait synchroniser directement les mouvements de notre bras et de notre main à celui du robot, via une transformation ou traduction (car DOF différent) décrit dans le descripteur sémantique.

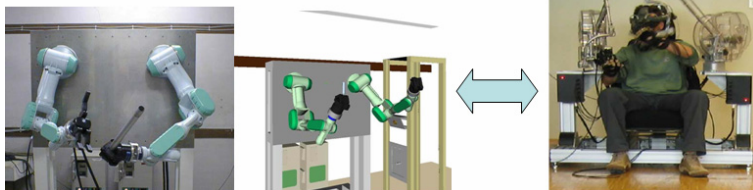


Figure 59 : Dans le concept Téléopérateur

les mouvements du bras de l'utilisateur sont coordonnés à ceux du robot. Il peut grâce à la « *station haptique* » ressentir les forces de contraintes.

## 5.5. CONCLUSION

Nous avons donc testé par le biais de ces applications notre concept de médiateur. Chacune des applications a apporté son lot de difficultés qui nous ont permis, petit à petit, d'améliorer nos interfaces, d'approfondir notre étude et de bien comprendre les problèmes, les difficultés et les attentes des utilisateurs. L'objectif final était de fournir une bonne interface homme machine, c'est à dire hautement configurable, générique, intuitive et permettant à l'utilisateur d'utiliser des concepts simples.





# CONCLUSION

## 6.1. RAPPELS

Le but de cette recherche a été de trouver une méthode permettant de manipuler des entités complexes plus facilement. Après avoir étudié de multiples techniques d'interactions et de manipulations, nous nous sommes dirigés peu à peu vers notre concept du médiateur. Un utilisateur via une interface de commande, construite dans un environnement virtuel 3D, va pouvoir agir au moyen d'un médiateur sur l'entité complexe (robot, objets virtuels) qu'il désire contrôler

**Les points particuliers suivants ont été mis en évidence :**

- ✓ L'utilisation simultanée de matériels très divers est source de nombreux problèmes de coexistence et de mise relation.
- ✓ Les temps de latence entre les différentes couches du système peuvent altérer la qualité de l'application, et provoquer un effet « *cybersickness* » pour les personnes sensibles.
- ✓ La synchronisation des tâches et calculs répartis sur plusieurs machines est parfois difficile.
- ✓ Les utilisateurs ont du mal, même sous une vision en mode stéréo, à estimer les distances entre leur instrument interactif (leur main) et l'objet médiateur qu'il désire manipuler.
- ✓ La qualité de l'environnement est insuffisante.

Malgré ces problèmes, les progrès constants des technologies vont faire que peu à peu, ils disparaîtront avec le temps.

## Les pistes d'améliorations envisagées ou envisageables :

### ✓ Pour accroître la visibilité des objets virtuels, il est possible d'utiliser :

- des ombres pour souligner les contours et augmenter le contraste,
- des changements de couleur en fonction de la position de l'objet par rapport à l'instrument interactif.
- des avertisseurs sonores

### ✓ Pour améliorer la qualité de l'environnement, il est possible :

- *de remplacer ou de transférer le stimulus d'une modalité par un stimulus d'une autre modalité.*

On peut donner une réaction « *haptique* » à l'utilisateur en déclenchant un son quand il touche un objet. [Biocca2001]. Mais, dans certains cas la combinaison multimodale dégrade le caractère proprioceptif en provoquant un conflit d'analyse cérébrale de l'information, comme cela se produit lorsqu'on essaye de descendre un escalier en colimaçon dans un système immersif CAVE.

- *de contrôler indirectement une entité du monde complexe par l'intermédiaire d'une main virtuelle agissant sur une entité médiatrice.*

La combinaison d'une analyse gestuelle du mouvement de l'utilisateur avec un retour d'effort et la mise en place d'indicateurs visuels semblent être un bon moyen pour résoudre les problèmes de détection de collisions.

- *d'accroître l'« affordance » du système par la pertinence des commandes.*

Il est plus simple de diriger un robot par le geste que de tourner un volant car cela demande trop de concentration. L'utilisateur peut contrôler plus facilement le robot si l'on associe les mouvements de son bras à un dispositif de retour d'effort qui le force à revenir en position d'attente.

- *d'accroître la sensation de présence dans notre environnement virtuel pour que la personne puisse se concentrer sur les tâches à accomplir.*

Le couplage de la perception et de l'action est une chose vraiment importante. De plus, lors de l'interaction, l'utilisateur doit pouvoir s'identifier à l'entité qu'il contrôle et faire corps avec elle.

Finalement, nous pouvons dire que la flexibilité, l'adaptabilité, l'« affordance », et la généricité sont des points clés incontournables qui font qu'une interface médiatrice soit une réussite ou non.

## 6.2. CONTRIBUTIONS

Les principales contributions de cette recherche sont :

- La création de la notion d'interface médiatrice ou plus simplement « *médiateur* ». Ce nouveau concept développé tout au long de mon écrit, a besoin d'être affiné en tenant compte des expérimentations futures.
- La création d'outils virtuels d'interaction ; outils médiateurs offrant une métaphore d'interaction pour la manipulation. Jusqu'à maintenant les expérimentations utilisant une « *station haptique* » pour manipuler et ressentir les objets, nous ont montré que le mode de fonctionnement classique pose quelques problèmes. Mais, le fait de passer par l'intermédiaire d'un médiateur (objet simple) pour pouvoir manipuler un objet complexe, améliore significativement les choses. L'utilisation d'une « *station haptique* » n'est pas à écarter de notre concept du médiateur. C'est un outil indispensable permettant à l'utilisateur de se concentrer sur la manipulation de son objet complexe via un objet médiateur (par exemple un joystick virtuelle)
- Mise en place au sein de notre environnement de techniques d'interactions pour favoriser la navigation, et l'interaction de l'utilisateur avec son monde virtuel. Comme par exemple l'implémentation de la métaphore du principe de la main virtuelle qui chauffe sous la forme de stimuli visuels et sonore lorsque celle-ci est à proximité d'un objet. Cette méthode facilite en effet l'estimation de la distance ainsi que la sélection.
- La proposition d'une sorte de télécommande virtuelle, facilement accessible et toujours disponible : le « *menu pétales* », permettant à l'utilisateur d'interagir sans difficultés dans son monde médiateur. Les diverses expérimentations nous ont montré, en effet, que ce système de menu est assez intuitif et s'est révélé comme être un élément indispensable pour que l'utilisateur puisse construire de manière conviviale sa

propre interface médiatrice. En effet par exemple, dans le cadre de la « téléopération », le « *menu pétales* » permet de choisir les outils nécessaires pour piloter un robot à distance.

- Ce que nous apportons par rapport au concept « *d'outils virtuels* » et aux travaux de recherche effectués par Gobbetti [Gobbetti1993], est de pouvoir par l'intermédiaire de notre interface médiatrice contrôler n'importe quelles entités complexes. Notre champ d'application ne se restreint pas uniquement à contrôler des objets complexes virtuels comme on le fait dans certaines applications de construction graphique, mais de fournir une interface de contrôle permettant par exemple dans le cadre de la téléopération contrôler des robots et... De plus les mécanismes internes de construction de l'interface sont différents. Un robot réel n'est toutefois pas nécessairement comparable à un objet virtuel 3D. Les mécanismes de contrôle sont différents.

### 6.3. L'INTERFACE MEDIATRICE ET SES POSSIBILITES D'UTILISATION

A partir des diverses expérimentations réalisées au cours de cette recherche, la structure générale d'une interface médiatrice à nos besoins, a pu être conceptualisée.

Pour construire cette interface nous avons dû cogiter et trouver des réponses concernant la structure et la gestion de l'environnement du médiateur tant au niveau adaptabilité qu'au niveau de la présentation des informations assurant sa fonctionnalité ;

Notre objectif était, de faciliter la perception de l'information et de permettre à l'utilisateur d'interagir avec elle sous forme d'une représentation et d'outils interactifs réels ou virtuels.

Nous avons mené des réflexions sur les impératifs de présentations visuelles qui ont abouti aux questionnements suivants :

- Comment augmenter la clarté et simplicité de l'utilisation de l'interface visuelle programmable ?

- Comment rendre l'information assimilable afin d'assurer une prise de décision rapide rendant ainsi l'interface plus interactive ?
- Comment trouver un équilibre entre ce que l'utilisateur peu assimiler et les moyens d'interactions ?
- Comment la répartition de l'information influe t-elle sur le comportement et freine-t-elle l'interactivité de l'interface ?
- Comment permettre à l'utilisateur de construire son environnement virtuel en utilisant des techniques de programmation visuelle, conviviales et simples à mettre en œuvre
- Comment définir clairement l'environnement de travail, les ressources et les besoins en tenant compte des contraintes de flux de données, afin de permettre à l'utilisateur d'interagir efficacement avec les objets virtuels : des entités médiatrices informatives ou interactives de type métaphorique ou idiomatique ?

Finalement, nous avons réussi à construire notre concept d'interface médiatrice composée de huit modules spécifiques.

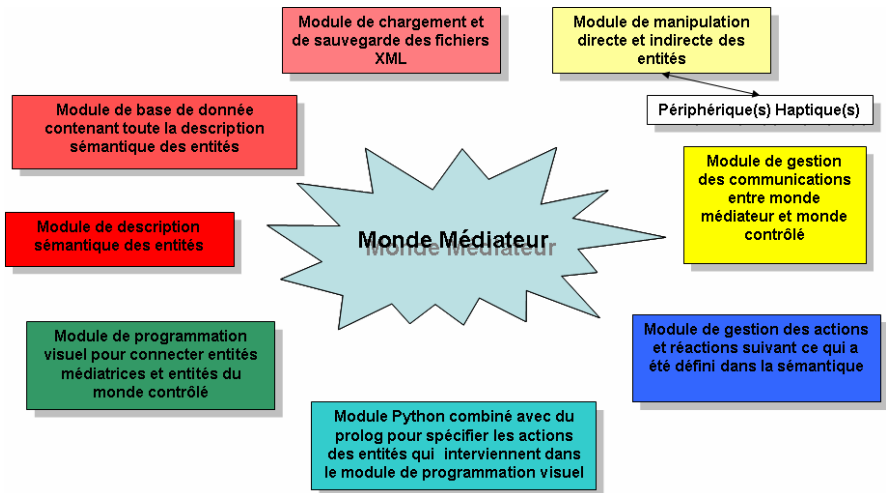


Figure : Diagramme de fonctionnement de monde médiateur composé de modules pour la plupart interconnectés.

## Brèves description des fonctionnalités des différents modules

- **Le module gérant les fichiers XML** fournirait toutes les informations nécessaires au module de la base de données. Ce module contiendrait tout les paramètres nécessaires (description sémantique des entités du monde contrôlé et du monde médiateur, description de l'interface médiatrice suivant le profil de la personne) et veille au bon fonctionnement de l'interface médiatrice.

- **Le module de programmation visuelle** permettrait la construction de l'interface médiatrice. Il comporterait de nombreux outils permettant d'agir sur les différentes entités, comme par exemple de mettre en relation la représentation symbolique d'une entité du monde contrôlé avec une entité du monde médiateur en respectant les règles imposées dans la description sémantique de chacun d'eux.

A noter que ces outils seraient aussi des entités dont les rôles seraient décrits dans le module python combiné avec du prolog. On pourrait par exemple appliquer des opérations sur deux entités du type : entité A + entité B donne entité C traduite par la suite de commandes « *combiner et remplacer* », qui serait présentée à l'utilisateur sous la forme d'une entité visuelle. Les opérations seraient prédéfinies dans des commandes du « *module python* » mais elles seraient réalisées par le « *module de manipulation* » qui gèrerait aussi les périphériques « *haptiques* ». L'utilisateur pourrait ainsi programmer visuellement son interface puis passer dans le mode simulation pour agir sur le monde contrôlé via les entités médiatrices, comme par exemple un joystick virtuel agissant sur un robot.

- **Le module de gestion des communications** transférerait dans les deux sens les données nécessaires, du monde médiateur vers le monde contrôlé par une connexion UDP.
- **Le module de description sémantique** permettrait de définir toutes sortes de périphériques hardware du monde réel afin de les connecter aisément au système.

Munie de ces modules, cette interface médiatrice serait hautement configurable parce que ses structures internes seraient flexibles car l'utilisateur via le module de programmation visuelle pourrait créer n'importe qu'elle interface simple ou complexe.

Après l'avoir sauvegardée, il pourrait la retrouver, l'utiliser ou la modifier à loisir pour l'ajuster à son profil.

Puisqu'elle permettrait de contrôler plusieurs entités du monde contrôlé, l'utilisateur pourrait donc changer entièrement son interface en fonction du type d'entité contrôlée.

La manipulation directe des périphériques de contrôle, virtuels et simples aurait pour avantages immédiats une certaine facilité d'apprentissage facile et de meilleures performances dans l'exécution des tâches.

#### 6.4. CONSTATATIONS ET RECOMMANDATIONS

Nous vous avons présenté dans les différents chapitres relatant les étapes de notre recherche, différents travaux qui ont permis la création de nouvelles techniques d'interaction et la modélisation d'une interface médiatrice.

Les nombreuses tentatives fructueuses ou non, et les expérimentations nous ont permis de progresser dans la conceptualisation du médiateur et nous en avons retiré beaucoup d'enseignements précieux. A titre de bilan, nous allons lister maintenant tous les éléments d'appréciations dont il faudrait tenir compte créer une bonne interface médiatrice.

##### ➤ **Sur le point de vue visuel il faudrait :**

- ✓ la qualité visuelle de l'interface pourrait être facilement amélioré par l'utilisation des mécanismes d'occultation, d'ombre, de perspective afin de pouvoir apprécier au mieux les distances.
- ✓ éviter les effets de miroirs ou de flashes qui pourraient distraire l'attention de l'utilisateur.
- ✓ garder un texte lisible avec une inclinaison inférieur à 30 degré en évitant l'uniformité pour augmenter la vigilance.
- ✓ faciliter la recherche de l'information en ayant recours à une organisation linéaire et à un arrangement des structures de sélection

➤ **Sur le point de vue interface, il faudrait :**

- ✓ simplifier les mouvements, la navigation de l'utilisateur et compenser l'immersion par le son ou l'« *haptique* », afin d'augmenter l'attention, l'implication et l'immersion de l'utilisateur
- ✓ simplifier au maximum les mouvements des objets qui doivent être contrôlés
- ✓ permettre à l'utilisateur d'enrichir son espace interactif ainsi que les actions en ayant la possibilité d'insérer ses nouveaux outils
- ✓ anticiper les besoins et désirs de l'utilisateur en lui proposant les outils et les informations nécessaires à l'accomplissement de sa tâche, pour lui éviter de les chercher
- ✓ réduire au maximum la charge cognitive de la mémoire à court terme. Les séquences d'actions devraient avoir un temps de vie limité et il faudrait produire un historique pour pouvoir revenir en arrière si c'est possible. Il est bien entendu que dans le cadre de la téléopération avec utilisation d'un robot, nous ne pouvons pas revenir en arrière car certaines actions irréversibles.
- ✓ prévenir et gérer les erreurs. (écrire des feedback messages compacts, design préventif, autosuggestion de l'autocorrectif) en faisant appel à un superviseur du système dont les règles de fonctionnement seraient préalablement définies par un expert

➤ **Sur le point de vue de la programmation et l'interaction visuelle, il faudrait :**

- ✓ faire correspondre le plus fidèlement possible, la forme des expressions visuelles utilisées pour la programmation ou l'interaction visuelle avec sa signification.
- ✓ minimiser la distance : le « *directness* » entre l'objectif et les actions à exécuter pour l'atteindre. Des outils interactifs simples doivent lui être fournis pour lui donner l'impression d'interagir directement sur les entités. Donc l'engagement, l'implication, la satisfaction sont des termes à prendre en considération pour tirer la meilleure partie de notre interface médiatrice.



## 6.5. CONCLUSION

Pour conclure cet écrit, nous pouvons signaler que beaucoup de recherches ont été faites dans le but de trouver de nouveaux outils interactifs et de créer de nouvelles structures dont la faisabilité a été prouvée. Mais que, jusqu'à ce jour, les outils caractéristiques du paradigme WIMP n'ont pas trouvé leurs homologues dans les interfaces à trois dimensions. Il reste encore du chemin à faire pour pouvoir trouver l'outil idéal et l'imposé comme un moyen standard pour les interfaces et applications 3D.

Notre objectif primaire était d'offrir à l'utilisateur des moyens efficaces pour pouvoir manipuler des objets complexes, mais malgré l'utilisation d'appareils « *haptiques* » pour faciliter cette interaction, les résultats nous ont montré que c'était loin d'être évident.

C'est pourquoi, nos recherches se sont focalisées vers une interaction indirecte de type menu 3D. Nos expériences successives, nous ont conduit à la création d'un autre type de structure : le « *menu pétales* » qui est un outil médiateur entre l'utilisateur et l'objet complexe. L'idée d'interface médiatrice a germé et nous a permis l'élaboration du nouveau « *concept médiateur* ».

Les premières expériences menées dans cette direction sont concluantes. Nous pensons avoir démontré par nos recherches les possibilités offertes par ce concept. Nous pensons qu'il serait bien de le développer de manière générale : interface conviviale et programmable par l'utilisateur, dotée d'outils simples, efficaces et puissants.

Nous pouvons répondre par l'affirmative à la question de savoir s'il est nécessaire ou non, de créer une interface médiatrice entre l'homme et l'entité complexe qu'il désire manipuler car c'est le seul moyen efficace qui permette de réaliser des interactions simples à mettre en œuvre.

L'homme a besoin d'outils simples pour pouvoir réaliser des tâches compliquées. Par principe, il faut se rappeler que ce n'est pas à l'homme de s'adapter à la machine, mais que c'est à la machine de s'adapter à lui.

---

# BIBLIOGRAPHIE

[3Dconnexion] 3Dconnexion.<http://www.3dconnexion.com/products>.

[Abel2000] P. Abel, P. Gros, C. R. D. Santos, D. Loisel, and J.-P. Paris. Automatic construction of dynamic 3d metaphoric worlds: An application to network management. In J. R. Robert Erbacher, Philip Chen and C. Wittenbink, editors, *Visual Data Exploration and Analysis VII*, volume 3960, pages 312–323. SPIE - The International Society for Optical Engineering, Jan. 2000.

[Angerilli2001] M. Angerilli, A. Frisoli, F. Salsedo, S. Marcheschi, M. Bergamasco, "Haptic simulation of an automotive manual gearshift", *Proceedings of Roman2001, International Workshop on Robot-Human Communication*. September 18-21 2001, Bordeaux-Paris, France.

[Baerlocher 2004] P. Baerlocher, R. Boulic, « An Inverse Kinematic Architecture Enforcing an Arbitrary Number of Strict Priority Levels », *The Visual Computer*, Springer Verlag, 20(6), 2004

[Bardinet1995] E. Bardinet, L.D. Cohen, and N. Ayache. Superquadrics and free-form deformations: a global model to fit and track 3D medical data. In *Proceedings of the Conference on Computer Vision, Virtual Reality and Robotics in Medecine (CVRMed'95)*, Lecture Notes in Computer Science, Nice, France, April 1995. Springer Verlag. Keyword(s): FFD, Superquadrique, Deformations, motion tracking. [bibtex-entry]

[Bardinet1995] Eric Bardinet, Laurent D. Cohen, et Nicolas Ayache. « A parametric deformable model to fit unstructured 3d data. » *Rapport de recherche 2617 - INRIA*, 1995.

- [Barr1981] Alan H. Barr. « Superquadrics et angle preserving transformations. » IEEE Computer Graphics and Applications, vol. 1, pp 11-23, Janvier 1981.
- [Biocca2001] Biocca, F., Kim, J. and Choi, Y. “Visual touch in virtual environments:An exploratory study of presence, multimodal interfaces, and cross-modal sensory illusions”, Presence:Teleoperators and Virtual Environments, 10(3), 247-265, 2001.
- [Bolt1980] Bolt, R., Put-that-there. SIGGRAPH 1980 proceedings, ACM press, pp. 262-270.
- [Both2001] S. Both, T.-Schmidt-Tjarksen (2001). Psychological Theory in Haptic Interfaces Design:Initial Steps Towards an Interactive Cognitive Subsystems (ICS) Approach. In Proceedings of EuroHaptics, 2001 Python (web). <http://www.python.org/>.
- [Boulic] R. Boulic, S. Rezzonico, D. Thalmann, Multi-Finger Manipulation of Virtual Objects, Proc. ACM VRST, ACM, pp.67-74.
- [Bowman1997] Doug A. Bowman. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In Symposium on Interactive 3D Graphics, 1997.
- [Bowman1998] D. Bowman, L. F. Hodges, and J. Bolter. The virtual venue:User-computer interaction in information-rich virtual environments. Presence:Teleoperation and Virtual Environments, 7(5):478–493, October 1998.
- [Bowman2001] D.A. Bowman, J. La Viola, M. Mine, and I. Pouyrev. Advanced tpoc in 3D user interface design. In Course Notes-SIGGRAPH 2001,2001
- [Burdea\_1\_1996] Burdea, G (1996) Force and touch feedback for virtual reality, New York:Wiley
- [Burdea\_2\_1996] Burdea, G., Richard, P. and Coiffet, P. (1996) Multimodal virtual reality:Input-output devices, system integration, and human factors. International Journal of Human-Computer Interaction. Vol. (1), 1996, pp. 5-24

[Burdea2003] BURDEA G. C.:Haptic feedback for virtual reality. In In Virtual Reality and PrototypingWorkshop (Laval (France), June 2003).

[Callahan1988] Callahan, J., D. Hopkins, M. Weiser, and B. Shneiderman, "" An Empirical Comparison of Pie and Linear Menus"", Human Factors in Computing Systems, SIGCHI Proceedings:ACM Press, 1988"

[Casier2003] CASIEZ G., PLÉNACOSTE P., CHAILLOU C., SEMAIL B.:The DigiHaptic, a new three degrees of freedom multi-finger haptic device. In Virtua Reality International Conference (May 2003), pp. 35.

[Chevennement1998] J. Chevennement. Malaises des opérateurs. Club CRINTransports, groupe Simulateurs d'études.

[Cohen1994] I. Cohen, L.D Cohen. – A hybrid hyperquadric model for 2-D and 3-D data fitting. – Rapport de recherche n 2188, INRIA, January 1994.

[Conner1992] Conner DB, Snibbe SS, Herndon KP, Robbins DC, Zeleznick RC, Van Dam A (1992) Three-Dimensional Widgets. SIGGRAPH Symposium on Interactive Graphics 183-188

[Darken1994] Darken, R.P., Hands-Off Interaction With Menus in Virtual Spaces. Proceedings of SPIE '94, Stereoscopic Displays and Virtual Reality Systems. Vol. 2177, pp. 365-371.

[Doug1997] Doug A. Bowman. An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments. In Symposium on Interactive 3D Graphics, 1997.

[Ellis1996] Ellis, S. R. "Presence of Mind:A Reaction to Thomas Sheridan's "Further Musings on the Psychophysics of Presence"", Presence:Teleoperators and Virtual Environments, 5(2), 247-259, 1996.

[Ferneau1995] M. Ferneau and J. Humphries. A gloveless interface for interaction in scientific visualization virtual environments. Proceedings of SPIE, Stereoscopic Displays and Virtual Reality Systems, 2409:268–274, 1995.

[Finch1995] Mark Finch, Mike Falvo, Vernon L. Chi, Sean Washburn, Russell M. Taylor II, and Richard Superfine. Surface Modification Tools in a Virtual Environment Interface to a Scanning Probe Microscope. In 1995 Symposium on Interactive 3D Graphics, pages 13-18,203. ACM SIGGRAPH, April 1995.

[Fisch2003] Fisch, A., Mavroidis, C., Melli-Huber, J., Bar-Cohen, Y.:Chapter 4:Haptic Devices for Virtual Reality, Telepresence, and Human-Assistive Robotics. In:Biologically-Inspired Intelligent Robots. SPIE Press (2003)

[Frisoli2001] Frisoli A, Avizzano CA, Bergamasco M, Data S, Santi C, "Dynamic modeling of primary commands for a car simulator", IEEE 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM '01) 8-12 July 2001 Como, Italy.

[Frisoli2002] Frisoli A, Simoncini F, Bergamasco M. (2002). Mechanical Design of a Haptic Interface for the Hand. ASME International DETC- 27th Biennial Mechanisms and Robotics Conference , Montreal- Canada, September 29 - October 2.

[Frisoli2005] A. Frisoli, F. Rocchi, S. Marcheschi, A. Dettori, F. Salsedo, and M. Bergamasco, "A New Force-Feedback Arm Exoskeleton for Haptic Interaction in Virtual Environments". World Haptics Conference (WHC 2005), March 18 - 20, 2005, Pisa, Italy.

[Fitts]. Fitts P.M., The information capacity of the human motor system in controlling the amplitude of the movement, Journal of experimental psychology 47,pp.381-391, 1954.

[Fong2001] Fong, T., Thorpe, C.:Vehicle teleoperation interfaces. In:Autonomous Robots.Volume 11., Kluwer Academic Publishers (2001)

[Fong2003] Fong, T., Thorpe, C., Glass, B.:PDADriver:A handheld system for remote driving. In:Proceedings of IEEE International Conference on Advanced Robotics, Coimbra, Portugal. (2003)

[Forsberg1996] Forsberg, A., Herndon, K., & Zeleznik, R. (1996). Aperture based selection for immersive virtual environments. Proceedings of the ACM Symposium on User Interface Software and Technology, 95--96.

[Fritz1999] Fritz, JP, and Barner, K.E. (1999). Design of haptic data visualization system for people with virtual impairments. IEEE Transactions on Rehabilitation Engineering, vol.7, no.3. Sept 1999, pp 372-84. Publisher:IEEE, USA

[Geoffrey1997] Geoffrey S. Hubona, Gregory W. Shirah, and David G. Fout 3d object recognition with motion. In Extended Abstracts of CHI'97, page 345-346, ACM, 1997.

[Gibson1977] Gibson, J. J. "The Theory of Affordances." In R. E. Shaw & J. Bransford (eds.), *Perceiving, Acting, and Knowing*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1977.

[Gibson1979] Gibson, J. J. *The Ecological Approach to Visual Perception*. Houghton Mifflin, Boston, 1979.

[Gilkey1995] Gilkey, R. H. and Weisenberger, J. M. "The sense of presence for the suddenly deafened adult", *Presence:Teleoperators and Virtual Environments*, 4(4), 357-363 1995.

[Gill1981] Gill, P. R.; Murray, W.; and Wright, M. H. ""The Levenberg-Marquardt Method."" §4.7.3 in *Practical Optimization*. London:Academic Press, pp. 136-137, 1981. "

[Gobbetti1993] E. Gobbetti, J.F. Balaguer, D.Thalmann, VB2: An Architecture for Interaction in Synthetic Worlds, Proc. ACM UIST '93, Atlanta, pp.167-178.

[Gomez1995] Gomez, D., Burdea, G., Lagrana, N. (1995). Integration of the Rutgers Master II in a virtual reality simulation. Proceedings of Virtual Reality Annual International Symposium '95. IEEE Computer Society Press, Los Alamitos, CA, pp. 198-202

[Gosselin2005] F. Gosselin, T. Jouan, J. Brisset, and C. Andriot. « Design of a Wearable Haptic Interface for Precise Finger Interactions in Large Virtual Environments". World Haptics Conference (WHC 2005), March 18 - 20, 2005, Pisa, Italy.

[Green2004] Green N, Kruger J., Faldu C., St. Amant R., A reduced QWERTY keyboard for mobile text entry, Proceedings of the ACM Conference on Human Factors in Computing Systems - CHI '04, pp. 1429- 1432, Vienne, 2004.

[Gregory2000] GREGORY A., EHMANN S., LIN M. C.:in-Touch:interactive multiresolution modeling and 3d painting with haptic interface. In Proceedings of IEEE International Conference on Virtual Reality 2000 (2000).

[Grojean2001] J.Grosjean and S. Coquillart. Command & Conrol Cube:a shortcut paradigm for virtual environments. In IPT-EGVE'2001, Stuttgart, Germany, May 2001

[Gruening1998] Gruening, J. et al (1998). Driving Simulation, SAE980223.

[Gutierrez-1-2004] M. Gutierrez, R. Ott, D. Thalmann, and F. Vexo. Mediators:Virtual haptic interfaces for tele-operated robots. In Proceedings of the 13th IEEE International Workshop on Robot and Human Interactive Communication (RO-MAN 2004), Kurashiki, Okayama, Japan, September 2004. pages 515-520

[Gutierrez-2-2004] M. Gutierrez, P. Lemoine, D. Thalmann, F. Vexo. Telerehabilitation: Controlling Haptic Virtual Environments through Handheld Interfaces. Proceedings of ACM Symposium on Virtual Reality Software and Technology (VRST 2004), November 2004, Hong Kong.

[Haiying2004] Calibrating Human Hand for Teleoperating the HIT/DLR Hand Haiying Hu, Xiaohui Gao, Jiawei Li, Jie Wang, and Hong Liu (ICRA '04) Proceedings of the 2004 IEEE International Conference on Robotics and Automation, vol.5, pp.4571- 4576.

[Hashimoto1993] H. Hashimoto, Y. Kunii, M. Buss, and F. Harashima, "Dynamic force simulator for force feedback human-machine interaction," in Proc. IEEE Virtual Reality Annual Int. Symp., Seattle, WA, 1993.

[Hendrix1996] Hendrix, C. and Barfield, W. "The sense of presence within auditory virtual environments", Presence:Teleoperators and Virtual Environments, 5(3), 290-301 1996.



[Hoffman1996] Hoffman, H., Groen, J., Rousseau, S.,Hollander,A. Win, W.,Wells, M., Furness, T.(1996). Tacticle augmentation:Enhancing presence in virtual reality with tactile feedback from objects, Paper presented at the meeting of American Psychology Society, 1996, San Francisco. CA

[Howe1992] R. D. Howe, “A force-reflecting teleoperated hand system for the study of tactile sensing in precision manipulation,” in Proc.IEEE Int. Conf. Robotics Automation, Nice, France, May 1992, pp. 1321–1326.

[Hudson1997] Thomas C. Hudson and Ming C. Lin and Jonathan Cohen and Stefan Gottschalk and Dinesh Manocha. “V-COLLIDE: accelerated collision detection for VRML”. ACM Press 1997,New York, NY, USA.

[Ichikawa1987] T. Ichikawa and M. Hirakawa. Visual programming, toward realization of user friendly programming environments. In Proceedings of the 1987 Fall Joint Computer Conference on Exploring technology:today and tomorrow, pages 129–137. IEEE Computer Society Press, 1987.

[ImmersionCorporation2003] Immersion Corporation:(Haptic workstation.  
<http://www.immersion.com> )

[InterSense] InterSense Co “IS600 Mark 2 Overview,” Burlington MA. Electronic version:[www.isense.com/is600.htm](http://www.isense.com/is600.htm).

[Iwata1992] H. Iwata, T. Nakagawa, and T. Nakashima, “Force display for presentation of rigidity of virtual objects,” J. Robot. Mechatron., vol. 4, no. 1, pp. 39–42, 1992.

[Johnson1995] A. Johnson and F.Fotouhi. SANDBOX Scientists Accessing Necessary Data Based On eXperimentation. In Interaction,2.3,34-45,86,1995

[Kallmann2002] Marcelo Kallmann and Daniel Thalmann (2002), Direct 3D Interaction with Smart Objects. Technical Report of Computer Graphics Lab, Swiss Federal Institute of Technology.

[Kallmann2003] Kallmann, M., Lemoine, P., Thalmann, D., Cordier, F., Magnenat-Thalmann, N., Ruspa, C., Quattrocchio, S. (2003). Immersive vehicle Simulators for Prototyping, Training and Ergonomics. Proceedings in Computer Graphics International CGI-03, Japan.

[Kalra1989] Devendra Kalra and Alan H. Barr. Guaranteed ray intersections with implicit surfaces. In Jeffrey Lane, editor, Computer Graphics (SIGGRAPH '89 Proceedings), volume 23, pages 297–306, July 1989.

[Kawasaki2003] KAWASAKI H., TAKAI J., TANAKA Y., MRAD C., MOURI T.:Control of multi-fingered haptic interface opposite to human hand. In Proceedings of the International Conference on Intelligent Robots and Systems (Las Vegas, 2003).

[Kim] N. Kim, G.J. Kim, C-M. Park, I. Lee, S.H. Lim. Multimodal Menu Presentation and Selection in Immersive Virtual Environments. Pohang University of Science and Technology, Korea.

[Krauss2001] Krauss, M., Scheuchenpflug, R., Piechulla, W. & Zimmer, A. (2001). Measurement of presence in virtual environments. In A. Zimmer, K. Lange, K.-H. Bäuml, R. Scheuchenpflug, R. Loose, O. Tucha, R. Findl & C. Schneider (Hrsg.), Experimentelle Psychologie im Spannungsfeld von Grundlagenforschung und Anwendung - Tagung experimentell arbeitender Psychologen (TeaP 2001) [CD-ROM] (S. 358-362). Regensburg: Universitätsbibliothek.

[Kurtenbach1993] Kurtenbach, G. and Buxton, W. (1993). The limits of expert performance using hierarchic marking menus. In CHI '93, pages 482–487, Amsterdam, The Netherlands. ACM Press. 2.1.4, 2.3, 9.2

[Kurtenbach1999] Kurtenbach, G., , Fitzmaurice, G. W., Owen, R. N., and Baudel, T. (1999). The hotbox:efficient access to a large number of menu-items. In CHI '99, pages 231–237, Pittsburgh PA, USA. ACM Press. 2.1.4

[LabVIEW] National Instruments Corporation. LabVIEW:Graphical development environment for signal acquisition, measurement analysis, and data presentation. <http://www.ni.com/labview/>.

[Larsen2000] Larsen, E., Gottschalk, S., Lin, M. C., and Manocha, D. (2000). Fast distance queries with rectangular swept sphere volumes.

[LaViola2000] J. LaViola. A discussion of cybersickness in virtual environments. ACM SIG-CHI Bulletin, 32 :47-56, 2000. ISBM 0736-6906.

[Lemoine2003] P. Lemoine, F. Vexo and D. Thalmann, Interaction Techniques: 3D Menus-based Paradigm, AVIR 2003

[Lemoine-1-2004] P. Lemoine, M. Gutiérrez, D. Thalmann, F. Vexo, The “Caddie Paradigm”: a free-Lo-motion Interface for Teleoperation, Joint workshop on CAPTECH and ENACTIVE: Believability in the virtual environment.

[Lemoine-2-2004] P. Lemoine, M. Gutierrez, F. Vexo, D. Thalmann, Mediators: Virtual Interfaces with Haptic Feedback, In Proceedings of EuroHaptics 2004, 5th-7th June, Munich, Germany, pages 68-73, 2004.

[Leonardis1994] Ales Leonardis, Franc Solina, et Alenka Macerl. « A direct recovery of superquadric models in range images using recover-and select paradigm. », Proceedings of 3d European Conference on Computer Vision, Stokholm, Mai 1994.

[Leonardis1997] Ales Leonardis, Ales Jaklic, et Franc Solina. « Superquadrics for segmenting and modeling range data. » IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(11), pp 1289-1295, Novembre 1997.

[Lessiter2001] Lessiter, J., Freeman, J., Keogh, E. and Davidoff, J. “A cross-media presence questionnaire:The ITC-sense of presence inventory”, Presence:Teleoperators and Virtual Environments, 10(3), 282-297, 2001.

- [Liang1994] Liang, J., Green, M., JDCAD:A highly interactive 3D modeling system. Computer & Graphics, 1994. 4(18):pp. 499-506.
- [Lindeman1999] R. Lindeman, J. Sibert, and J. Hahn. Hand-held windows :Towards effective 2D interaction in immersive virtual environments. In IEEE VR'99,1999
- [Lindeman1999] R. Lindeman, R.W., Sibert, J.L. and Hahn, J.K. (1999). Towards usable VR:An empirical study of user interfaces for immersive virtual environments. Proceedings of the Conference on Human Factors in Computing Systems CHI '99/ACM, 64-71, Pittsburgh, PA.
- [Lombard1997] Lombard, M. and Ditton, T. "At the Heart of It All:The Concept of Presence", Journal of Computer Mediated Communication, 3(2), 1997.
- [Manitiu2000] Abndrei Manitiu, interface 3D dans les mondes virtuels :proposition d'interaction générique, université Paul Sabatier, Toulouse, octobre 2000.
- [Mark1997] Mark R. Mine, Frederick P. Brooks, Jr., and Carlo H. Sequin. Moving objects in space:Exploiting proprioception in virtual-environment interaction. SIGGRAPH 97 Conference Proceedings, pages 19--26. 1997.
- [Massie1994] MASSIE T. H., SALISBURY J. K.:The phantom haptic interface:A device for probing virtual objects. In Proceedings of ASME Winter Annual Meeting Symposium. Haptic Interfaces forVirtual Environment and Teleoperator Systems(1994).
- [McGuffin2002] Michael J. McGuffin, Nicholas Burtnyk, Gordon Kurtenbach:FaST Sliders:Integrating Marking Menus and the Adjustment. Graphics Interface 2002:35-42
- [Mendoza2001] MENDOZA C., LAUGIER C.:Realistic haptic rendering for highly deformable virtual objects. In Proceedings of IEEE International Conference on Virtual Reality 2001 (2001), pp. 264–269.

[Miller 2004] Andrew T. Miller, Steffen Knoop, Peter K. Allen, Henrik I. Christensen. ""Automatic grasp planning using shape primitives."" In Proceedings of the IEEE International"

[Mine1997] Mark R.Mine, Frederick P. Brooks, Jr., and Carlo H.Séquin. Moving objects in space:exploiting proprioception in virtual environment interaction. In Turner Whitted, editor, SIGGRAPH97 Conference Proceedings, Annual Conference Series, pages 19-26,ACM SIGGRAPH, Addison Wesley, August 1997, ISBN0-98791-896-7

[Mine1997] Mark R.Mine, M., Brooks, F., and Sequin, C. (1997) Moving Objects in Space:Exploiting Proprioception in Virtual-Environment Interaction. Proceedings of SIGGRAPH, in Computer Graphics, 19-26.

[Mirtich1998] Mirtich, B. (1998). V-clip :fast and robust polyhedral collision detection. ACM Transactions on Graphics (TOG), 17(3) :177.208.

[Newman1973] Newman, W.M. and R.F. Sproull, Principles of Interactive Computer Graphics, 1st. edition, Berkeley, CA:McGraw-Hill, 1973 (discusses circular menus in general, but not Pie Menus)

[Nguyen2001] Nguyen, L.A., Bualat, M., Edwards, L.J., Flueckiger, L., Neveu, C., Schwehr, K., Wagner, M.D., Zbinden, E.:Virtual reality interfaces for visualization and control of remote vehicles. In:Autonomous Robots. Volume 11., Kluwer Academic Publishers (2001)

[Nitsche2001] Nitzsche, N., Hanebeck, U., Schmidt, G.:Mobile haptic interaction with extended real or virtual environments. In:Proceedings of 10th IEEE International Workshop on Robot and Human Interactive Communication. (2001) 313{318

[O'Toole1997] R. O'Toole, R. Playter, W. Blank, N. Cornelius, W. Roberts,and M. Raibert. A Novel Virtual Reality Surgical Trainer with Force Feedback:Surgeon vs. Medical Student Performance. In J. K. Salisbury and M. A. Srinivasan, editors, The Proceedings of the Second PHANTOM Users Group Workshop. December 1997.

[Ott2005] R. Ott, M. Gutierrez, D. Thalmann, F. Vexo, Improving User Comfort in Haptic Virtual Environments through Gravity Compensation, Proceedings of the First Joint Eurohaptics Conference And Symposium On Haptic Interfaces For Virtual Environment And Teleoperator Systems :WorldHaptics'05, 18-20 Mars, Pise, 2005

[Paljic2002] A. Paljic, J. M. Burkhardt, and S. Coquillart. A study of manipulation on three responsive workbench. In IPT'2002 Symposium, Orlando, US, 2002

[Park1992] Park, T. (1992). Design and analysis of Ford driving simulator, In Proceedings of IMAGE, Phoenix, USA.

[Pierce1997] J.S. Pierce, A.S. Forsberg, M.J. Conway, S.P. Hong, R.C. Zeleznik, and M.R. Mine, ""Image plane interaction techniques in 3D immersive environments"". In Proceedings of 1997 Symposium on Interactive 3D Graphics."

[Pierce1999] Jeffrey S. Pierce, Brian C. Stearns, Randy F. Pausch:Voodoo dolls:seamless interaction at multiple scales in virtual environments. SI3D 1999:141-145

[Pinho2002] Pinho, M.,Bowman, D. Freitas, C. Cooperative Object Manipulation in Immersive Virtual Environments:Framework and Techniques. Proceedings of ACM Virtual Reality Software and Technology, 2002, pp. 171-178

[Poupyrev1996] Poupyrev, I., Billingham, M., Weghorst, S., and Ichikawa, T. Go-Go Interaction Technique:Non-Linear Mapping for Direct Manipulation in VR. Proceedings of UIST 1996, pages 79-80.

[Reggiani2002] Reggiani, M., Mazzoli, M., and Caselli, S. (2002). An Experimental Evaluation of Collision Detection Packages for Robot Motion Planning. In IEEE International Conference on Intelligent Robots and Systems, IROS'02, Lausanne, Switzerland.

[Riecke2002] Riecke, B.E. and von der Heyde, M. "Qualitative Modeling of Spatial Orientation Processes using Logical Propositions:Interconnecting Spatial Presence, Spatial Updating, Piloting, and Spatial Cognition.", Technical report n. 100, Max Planck Institute for Biological Cybernetics, Tübingen, Germany, 2002.

[Rollo1992] Rollo, C., ""Pie Menus for Windows"", Dr. Dobb's Journal, Nov., 1992, pp 30-39"

[Ruspa\_1\_2002] Dymott, R., Harke, S., Scheuchenpflug, R., Ruspa, C., & Quattrocolo, S. Older drivers in virtual reality assessments of vehicle ergonomics. In D. de Waard, K.A. Brookhuis, S.M. Breker, and W.B. Verwey, Human Factors in the Age of Virtual Reality. Maastricht, the Netherlands: Shaker Publishing.

[Ruspa\_2\_2002] Ruspa, C., Scheuchenpflug, R. & Quattrocolo, S. Validity of virtual reality driving simulators for ergonomic assessment. In D. de Waard, K.A. Brookhuis, S.M. Breker, and W.B. Verwey, Human Factors in the Age of Virtual Reality. Maastricht, the Netherlands: Shaker Publishing.

[Ruspa\_3\_2002] Scheuchenpflug, R., Ruspa, C. & Quattrocolo, S. (in press). Presence in virtual driving simulators. In D. de Waard, K.A. Brookhuis, S.M. Breker, and W.B. Verwey, Human Factors in the Age of Virtual Reality. Maastricht, the Netherlands: Shaker Publishing.

[Sahling2002] Nikolaus Sahling, from high-dimensional data to insight. <http://www.vrvis.at/vis/resources/DA-NSahling/nodel.htm>

[Salisbury1997] J.K. Salisbury, C. Tarr, Haptic Rendering of Surfaces Defined by Implicit Functions, Proc. Sixth Annual Symp. on Haptic Interfaces for virtual Environment and Teleoperator Systems, ASME, Fairfield, N.J., 1997.

[Scheuchenpflug2002] Scheuchenpflug, R., Quattrocolo, S. & Ruspa, C. (2002). Messung der wahrgenommenen Qualität von Fahrsimulationen. In M. Baumann, A. Keinath & J. Krems (Hrsg.), Experimentelle Psychologie. Abstracts der 44 Tagung experimentell arbeitender Psychologen (S. 33). Regensburg:Roderer.

[Schuemie1999] M.J. Schuemie, C.A.P.G. van der Mast (1999). Presence:Interacting in VR?. In Proceedings of the TWLT 15, University of Twente, 1999, 213-217

[Snibbe2001] Snibbe, S.S., MacLean, K.E., Shaw, R., Roderick, J., Verplank, W.L., Schee®, M.:Haptic techniques for media control. In:Proceedings of the 14th annual ACM symposium on User interface software and technology, ACM Press (2001) 199{208}

[Solina1990] Franc Solina et Ruze<sup>2</sup>na Bajcsy. « Recovery of parametric models from range images:the case for superquadrics with global deformations. » IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(2), pp 131-147, Février 1990.

[Springer1997] Springer S.L. and Gadh, R. (1997). Haptic feedback for virtual reality computer aided design. Concurrent Product Design and Environmentally Conscious Manufacturing American Society of Mechanical Engineers,Design v94 1997, ASME, Fairfield, NJ, USA. P 1-8

[Springer1999] SPRINGER S., FERRIER N.:Design of a multifinger haptic interface for teleoperational grasping. In ASME Int'l Mech. Eng. Congress and Expo (November 1999).

[Srinivasan1997] M.A. Srinivasan, C. Basdogan, Haptics in Virtual Environments :Taxonomy, Research Status, and Challenges, Computers Graphics, Special Issue on Haptic Displays in Virtual Environments, Vol. 21, No.4, 1997.

[Stoakley1995] R. Stoakley, M. Conway and R. Pausch, ""Virtual reality on a WIM:interactive worlds in miniature"", Proceedings of CHI'95, pp. 265-272 (1995)"

[Stone1992] Stone, R. (1992) Virtual reality tutorial. MICAD Conference, Micado, Paris, France.

[Stone1993] Stone, R. J. 1993. Virtual Reality Systems. Earnshaw, R.A., Gigante, M. A., and Jones, H. (editors).. London, Academic Press.

[Strauss1992] Strauss PS, Carey R (1992) An Object-Oriented 3D Graphics Toolkit. Proc SIGGRAPH:341-347

[Tarrin2003]TARRIN N., COQUILLART S., HASEGAWA S., BOUGUILA L., SATO M.:The stringed haptic workbench:a new haptic workbench solution. In Proceedings of Eurographics (2003).



[Thomas1996] Thomas Harold Massie. Initial Haptic Explorations with the Phantom: Mrtual Touch Through Pointer Interaction. Master's thesis, Massachusetts Institute of Technology, February 1996.

[V-COLLIDE] V-Collide (web). [http:// www.cs.unc.edu /~geom / V-COLLIDE /](http://www.cs.unc.edu/~geom/V-COLLIDE/)

[Virtools] Virtools SA. Interactive 3d content production and development tools. <http://www.virtools.com>.

[Vodslay1997] D. Vodislav. A visual programming model for user interface animation. In Proceedings of IEEE Symposium on Visual Languages, 23-26 Sept., pages 344 – 351, 1997.

[Walker1990] Walker, N., & Smelcer, J.B. (1990). A Comparison of Selection Times from Walking and Pull-Down Menus. Proceedings of SIGCHI 1990, 221—225.

[Wang1999] Wang, Y. (1999). Object Transportation and Orientation in Virtual Environments. Ph.D. Thesis, School of Kinesiology, Simon Fraser University, Burnaby, BC, Canada.

[Zhao2004] Shengdong Zhao, Ravin Balakrishnan, symposium on User Interface Software and Technology Proceedings of the 17th annual ACM symposium on User interface software and technology Santa Fe, NM, USA SESSION: Gestures Pages:33 – 42 Year of Publication:2004 ISBN:1-58113-957-8

---

---

# CURRICULUM



Patrick Lemoine né le 19 mai 1970 à Besançon, France.

Il a obtenu un diplôme d'études avancées (DEA) d'Instrumentation et Informatique de l'Image à l'université de Bourgogne - Franche Comté en juillet 2000.

Depuis mars 2001, il travaille en tant qu'assistant de recherche au laboratoire d'Infographie (VRlab) à l'Ecole Polytechnique Fédérale de Lausanne.

## Publications

[Kallmann2003] Kallmann, M., Lemoine, P., Thalmann, D., Cordier, F., Magnenat-Thalmann, N., Ruspa, C., Quattrocchio, S. (2003). Immersive vehicle Simulators for Prototyping, Training and Ergonomics. Proceedings in Computer Graphics International CGI-03, Japan.

[Lemoine2003] P. Lemoine, F. Vexo and D. Thalmann, Interaction Techniques: 3D Menus-based Paradigm, AVIR 2003

[Gutierrez-2-2004] M. Gutierrez, P. Lemoine, D. Thalmann, F. Vexo. Telerehabilitation: Controlling Haptic Virtual Environments through Handheld Interfaces. Proceedings of ACM Symposium on Virtual Reality Software and Technology (VRST 2004), November 2004, Hong Kong.

---

[Lemoine-1-2004] P. Lemoine, M. Gutiérrez, D. Thalmann, F. Vexo, The “Caddie Paradigm”: a free-Loconotion Interface for Teleoperation, Joint workshop on CAPTECH and ENACTIVE: Believability in the virtual environment.

[Lemoine-2-2004] P. Lemoine, M. Gutierrez, F. Vexo, D. Thalmann, Mediators: Virtual Interfaces with Haptic Feedback, In Proceedings of EuroHaptics 2004, 5th-7th June, Munich, Germany, pages 68-73, 2004.

---

---